AFRL-IF-RS-TR-2000-117
Final Technical Report
August 2000


# WAVELENGTH DIVISION MULTIPLEXED NETWORKS WITH MINIMAL CONVERSION AND SWITCHING

University of Hawaii

Sponsored by
Defense Advanced Research Projects Agency
DARPA Order No. A490

*APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.*



AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2000-117 has been reviewed and is approved for publication.

APPROVED: Paul Sierak

PAUL SIERAK
Project Engineer

FOR THE DIRECTOR:

WARREN H. DEBANY, Technical Advisor
Information Grid Division
Information Directorate

If your address has changed or if you wish to be removed from the Air Force Research Laboratory Rome Research Site mailing list, or if the addressee is no longer employed by your organization, please notify AFRL/IFGA, 525 Brooks Road, Rome, NY 13441-4505. This will assist us in maintaining a current mailing list.

Do not return copies of this report unless contractual obligations or notices on a specific document require that it be returned.

# WAVELENGTH DIVISION MULTIPLEXED NETWORKS WITH MINIMAL CONVERSION AND SWITCHING

Galen H. Sasaki

Contractor: University of Hawaii
Contract Number: F30602-97-1-0342
Effective Date of Contract: 16 September 1997
Contract Expiration Date: 15 December 1999
Short Title of Work: Wavelength Division Multiplexed Networks with Minimal Conversion and Switching
Period of Work Covered: Sep 97 – Dec 99

Principal Investigator: Galen H. Sasaki
Phone: (808) 956-6103
AFRL Project Engineer: Paul Sierak
Phone: (315) 330-7346



This research was supported by the Defense Advanced Research Projects Agency of the Department of Defense and was monitored by Paul Sierak, AFRL/IFGA, 525 Brooks Road, Rome, NY.

| REPORT DOCUMENTATION PAGE | Form Approved OMB No. 0704-0188 |
|---|---|



| 1. AGENCY USE ONLY *(Leave blank)* | 2. REPORT DATE | 3. REPORT TYPE AND DATES COVERED |
|---|---|---|
| | AUGUST 2000 | Final Sep 97 - Dec 99 |

**4. TITLE AND SUBTITLE**
WAVELENGTH DIVISION MULTIPLEXED NETWORKS WITH MINIMAL CONVERSION AND SWITCHING

**5. FUNDING NUMBERS**
C - F30602-97-1-0342
PE - 61101F
PR - A490
TA - 00
WU - 01

**6. AUTHOR(S)**
Galen H. Sasaki

**7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)**
University of Hawaii
Department of Electrical Engineering
2540 Dole Street
Honolulu HI 96822

**8. PERFORMING ORGANIZATION REPORT NUMBER**
N/A

**9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)**
Defense Advanced Research Projects Agency
3701 North Fairfax Drive
Arlington VA 22203-1714

Air Force Research Laboratory/IFGA
525 Brooks Road
Rome NY 13441-4505

**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**
AFRL-IF-RS-TR-2000-117

**11. SUPPLEMENTARY NOTES**
Air Force Research Laboratory Project Engineer: Paul Sierak/IFGA/(315) 330-7346

**12a. DISTRIBUTION AVAILABILITY STATEMENT**
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

**12b. DISTRIBUTION CODE**

**13. ABSTRACT** *(Maximum 200 words)*
Optical network architectures that employ wavelength division multiplexing are proposed and evaluated at two layers. The first layer dynamically switches optical connectors. Network architectures at this layer allow a limited amount of wavelength switching by the optical connections. Network topologies and efficient resource allocation algorithms are proposed and evaluated. The second layer dynamically switches electronic traffic streams and multiplexes them onto the optical connections. network architectures at this layer provide maximal switching capability, but at significantly reduced equipment costs.

**14. SUBJECT TERMS**
Communications, Fiber Optics, Wavelength Division Multiplexing, Modeling

**15. NUMBER OF PAGES**
204

**16. PRICE CODE**

| 17. SECURITY CLASSIFICATION OF REPORT | 18. SECURITY CLASSIFICATION OF THIS PAGE | 19. SECURITY CLASSIFICATION OF ABSTRACT | 20. LIMITATION OF ABSTRACT |
|---|---|---|---|
| UNCLASSIFIED | UNCLASSIFIED | UNCLASSIFIED | UL |

Standard Form 298 (Rev. 2-89) (EG)
Prescribed by ANSI Std. 239.18
Designed using Perform Pro, WHS/DIOR, Oct 94

# Abstract

Optical network architectures that employ wavelength division multiplexing are proposed and evaluated at two layers. The first layer dynamically switches optical connections. Network architectures at this layer allow a limited amount of wavelength switching by the optical connections. Network topologies and efficient resource allocation algorithms are proposed and evaluated. The second layer dynamically switches electronic traffic streams and multiplexes them onto the optical connections. Network architectures at this layer provide maximal switching capability, but at significantly reduced equipment costs.

# Contents


**1 Summary** 1

**2 Introduction** 3

2.1 Limited Wavelength Conversion Networks . . . . . 6

2.1.1 Channel Attachment Patterns (CAPs) . . . . . 9

2.1.2 Channel Assignment for Pre-Computed Routes and No Rearrangement . . . . . 15

2.1.3 Channel Assignment With Pre-Computed Routes and Rearrangement . . . . . 18

2.1.4 Channel Assignment Without Pre-Computed Routes . . . . . 24

2.2 Networks With Minimal ADM/DCSs . . . . . 28

2.3 Organization . . . . . 30

**3 Methods, Assumptions, and Procedures** 32

3.1 Limited Wavelength Conversion Networks . . . . . 33

3.1.1 Network Model . . . . . 33

3.1.2 Traffic Model . . . . . 37

3.1.3 Simulation Details . . . . . 40

3.1.4 Simulation Implementation . . . . . 42

3.2 Networks with Minimal ADM/DCSs . . . . . 42

3.2.1 Network Model . . . . . 43

3.2.2 Traffic Models . . . . . 46

**4 Results and Discussion** 51

4.1 Limited Wavelength Conversion Networks . . . . . 51

4.1.1 Simple Channel Assignment . . . . . 51

4.1.2 Channel Assignment Patterns (CAPs) . . . . . 61

4.1.3 Channel Assignment With Rearrangement . . . . . 69

4.1.4 Channel Assignment Without Pre-Computed Routes . . . . . 73

4.1.5 DesignCAP . . . . . 82

4.2 Networks with Minimal ADMs/DCSs . . . . . 86

4.2.1 Rings . . . . . 87

4.2.2 General Topologies . . . . . 93

**5 Conclusions** 102

**A Channel Assignment Simulations for NSFNET** **110**

**B Channel Attachment Comparisons** **117**

**C Cost Effective Traffic Grooming in WDM Rings** **125**
C.1 Introduction . . . . . 126
C.1.1 Design assumptions and approach . . . . . 128
C.1.2 Traffic models . . . . . 130
C.1.3 Proposed network architectures . . . . . 131
C.2 Optical WDM Ring Architectures . . . . . 133
C.2.1 Fully-Optical Ring . . . . . 133
C.2.2 Single-Hub Ring . . . . . 135
C.2.3 Double-Hub Ring . . . . . 136
C.2.4 Point-to-Point WDM Ring . . . . . 139
C.2.5 Hierarchical Ring . . . . . 139
C.2.6 Incremental Ring . . . . . 141
C.3 Comparisons . . . . . 146
C.4 Conclusions . . . . . 153

**D A WDM Ring Network for Incremental Traffic** **156**
D.1 Introduction . . . . . 156
D.2 Incremental Ring . . . . . 160
D.3 Simulation Results . . . . . 165
D.4 Conclusions . . . . . 169

**E A Minimal Cost WDM Network for Incremental Traffic** **170**
E.1 INTRODUCTION . . . . . 170
E.2 NETWORK AND TRAFFIC MODELS . . . . . 172
E.3 NONBLOCKING RESULTS . . . . . 175
E.4 A SIMPLE DESIGN ALGORITHM . . . . . 178
E.5 SUMMARY . . . . . 180

**F List of Symbols, Abbreviations, and Acronyms** **183**

# List of Figures


2.1 A node in a point-to-point network. . . . . 4
2.2 A network with two lightpaths $L1$ and $L2$. . . . . 5
2.3 A node in a wavelength routed network. . . . . 6
2.4 Examples of wavelength conversion. . . . . 7
2.5 PARTITION for $d = 2$ and 4, and $W = 8$. . . . . 10
2.6 SHIFTED for $d = 2$ and 4, and $W = 8$. . . . . 11
2.7 DISTRIBUTE for $d = 2$ and 4, and $W = 8$. . . . . 11
2.8 SHUFFLE for $d = 2$ and 4, and $W = 8$. . . . . 13
2.9 A CAP and one of its branch exchanges. . . . . 14
2.10 Algorithm DesignCAP. . . . . 14
2.11 Channels along a lightpath's route. . . . . 16
2.12 Rearrangement-after-shutdown (RAS) algorithm. . . . . 20
2.13 Single move-to-vacant (MTV) algorithm. . . . . 21
2.14 Concept of sublightpath and maximal sublightpath. . . . . 22
2.15 MTV sublightpaths and their corresponding crossover edges. . . . . 23
2.16 Multiple move-to-vacant (MTV) algorithm. . . . . 25
2.17 A node in a ring network. . . . . 29

3.1 Ring with 16 nodes. . . . . 34
3.2 NSFNET. . . . . 35
3.3 Another Real Network Topology (REALNET) . . . . . 35
3.4 The WORST-CASE topology . . . . . 36
3.5 A Randomly Generated Topology (RANDOM) . . . . . 36
3.6 An opaque node. . . . . 45
3.7 An optical node. . . . . 45

4.1 Blocking probabilities for the 16 node ring, w-degree 2, DFS, and PARTITION. . . . . 53
4.2 Blocking probabilities for the 16 node ring, w-degree 2, DFS, and SHIFTED. 54
4.3 Blocking probabilities for the 16 node ring, w-degree 2, DFS, and DISTRIBUTE. . . . . 55
4.4 Blocking probabilities for the 16 node ring, w-degree 2, DFS, and RANDOM-SAME. . . . . 56
4.5 Blocking probabilities for the 16 node ring, w-degree 2, DFS, and RANDOM-RANDOM. . . . . 57

4.6 Blocking probabilities for the 16 node ring, w-degree 2, DFS, and SHUFFLE. 58
4.7 Blocking probabilities for the ring and w-degree 2. . . . . 59
4.8 Blocking probabilities for the ring and w-degree 4. . . . . 60
4.9 Blocking probabilities for NSFNET and w-degree 2. . . . . 61
4.10 Blocking probabilities for the 16 node ring, w-degree 2, and DFS with Fixed. . . . . 63
4.11 Blocking probabilities for the 16 node ring, w-degree 2, and DFS with Pack. 64
4.12 Blocking probabilities NSFNET, w-degree 2, and Fixed. . . . . 65
4.13 Blocking probabilities for NSFNET, w-degree 2, and Pack. . . . . 66
4.14 Blocking probabilities for the 16 node ring, Fixed, and PARTITION. . . 67
4.15 Blocking probabilities for the 16 node ring, Fixed, and DISTRIBUTE. . 68
4.16 Blocking probabilities for NSFNET, Fixed, and PARTITION. . . . . 68
4.17 Blocking probabilities for NSFNET, Fixed, and DISTRIBUTE. . . . . 69
4.18 Comparison of Channel Assignment with Rearrangement for the Ring with 16 wavelengths . . . . . 70
4.19 Comparison of Channel Assignment with Rearrangement for the Ring with 32 wavelengths . . . . . 71
4.20 Comparison of Channel Assignment with Rearrangement for the NSFNET with 16 wavelengths . . . . . 72
4.21 Comparison of Channel Assignment with Rearrangement for the NSFNET with 32 wavelengths . . . . . 72
4.22 Comparison of Algorithms — REALNET Topology with Full Conversion 74
4.23 Comparison of Algorithms — REALNET Topology with No Conversion . 75
4.24 Comparison of Algorithms — REALNET Topology with DISTRIBUTE (w-degree 4) . . . . . 76
4.25 Comparison of Algorithms — RANDOM Topology with No Conversion . 76
4.26 Comparison of Algorithms — NSFNET Topology with Full Conversion . 77
4.27 Plot for Brute Force Algorithm — WORST-CASE Topology, DISTRIBUTE CAP. . . . . 78
4.28 Plot for Brute Force Algorithm — RANDOM Topology, DISTRIBUTE CAP. . . . . 78
4.29 Performance variation with w-degree — Widest Shortest Algorithm on WORST-CASE Topology, DISTRIBUTE CAP. . . . . 79
4.30 Performance variation with w-degree — Shortest Distance Algorithm on WORST-CASE Topology, DISTRIBUTE CAP. . . . . 80
4.31 Comparison of Algorithms — WORST-CASE topology with PARTITION CAP and w-degree 4 . . . . . 80
4.32 Performance variation with degree of conversion — Brute Force Algorithm on WORST-CASE Topology with PARTITION CAP. . . . . 81
4.33 Performance variation with degree of conversion — Shortest Distance Algorithm on WORST-CASE Topology with PARTITION CAP. . . . . 81
4.34 Performance variation with degree of conversion — Widest Shortest Algorithm on WORST-CASE Topology with PARTITION CAP . . . . . 82

4.35 Comparison of Attachment Patterns — WORST-CASE topology and Shortest Distance Algorithm . . . . . 83
4.36 Comparison of Attachment Patterns — WORST-CASE topology and Shortest Distance Algorithm . . . . . 83
4.37 Results of DesignCAP from different starting CAPs. . . . . 84
4.38 The CAPs found by DesignCAP starting from DISTRIBUTE'. . . . . 85
4.39 The CAP SHUFFLE and the final CAP found by DesignCAP starting from SHUFFLE. . . . . 86
4.40 A section of a ring with $W = 8$. On the left is the section when is a part of a point-to-point network. On the right is when nodes are deleted from its transit wavelengths. . . . . 89
4.41 The ratio $ADMAVG/W$ as a function of $F$ for $N = 8, 12$, and 16. . . . . 90
4.42 Average WDM channel utilization for a ring network with $N = 8$, $W = 8$, $g = 4$, and $t(i) = 4$ for all nodes $i$. The average channel utilization is a function of $log_{10}(a)$. For this plot, the value of $a$ ranges from 0.1 to 10.0. 92
4.43 Distribution of the lengths of traffic streams that are accepted by the ring network with $N = 8$, $W = 8$, $g = 4$, and $t(i) = 4$ for all nodes $i$. . . . . 92
4.44 Average WDM channel utilization for a ring network with $N = 16$, $W = 4$, $g = 2$, and $t(i) = 1$ for all nodes $i$. The average channel utilization is a function of $log_{10}(a)$. For this plot, the value of $a$ ranges from 0.1 to 10.0. 94
4.45 Fraction of conforming traffic streams that were blocked in a ring network with $N = 16$, $W = 4$, $g = 2$, and $t(i) = 1$ for all nodes $i$. The average channel utilization is a function of $log_{10}(a)$. For this plot, the value of $a$ ranges from 0.1 to 10.0. . . . . 95
4.46 A General Topology Optical Network. . . . . 96
4.47 Table showing the average minimum cost for different ranges of values of $\{t(i)\}$. . . . . 96
4.48 Optical network . . . . . 98
4.49 Blocking rates ($MHT$ = Mean Holding Time) . . . . . 101

A.1 NSFNET, w-degree 2, PARTITION. . . . . 111
A.2 NSFNET, w-degree 2, SHIFTED. . . . . 112
A.3 NSFNET, w-degree 2, DISTRIBUTE. . . . . 113
A.4 NSFNET, w-degree 2, and RANDOM-SAME. . . . . 114
A.5 NSFNET, w-degree 2, and RANDOM-RANDOM. . . . . 115
A.6 NSFNET, w-degree 2, and SHUFFLE. . . . . 116

B.1 Blocking probabilities for the 16 node ring, w-degree 4, and FIXED. . . 118
B.2 Blocking probabilities for the 16 node ring, w-degree 4, and PACK. . . . 119
B.3 Stars 1 and 2. . . . . 119
B.4 Tree 1. . . . . 120
B.5 Tree 2. . . . . 120
B.6 The Mesh topologies for 16 wavelengths. . . . . 121
B.7 Random 1. . . . . 122
B.8 Random 2. . . . . 122

B.9 Mesh connected rings. . . . . 123
B.10 The mesh topologies for 16 wavelengths. . . . . 124

C.1 An optical WDM ring. . . . . 126
C.2 An optical node. . . . . 127
C.3 A point-to-point OADM ring with three wavelengths. . . . . 127
C.4 Setting up a lightpath between the first two nodes. . . . . 134
C.5 Setting up the lightpaths for two new nodes. . . . . 134
C.6 A single hub network for the case when $t_A(i) = 1$ for all nodes $i$. . . . . 135
C.7 A double-hub network when $t_A(i) = 2$ for all nodes $i$. . . . . 136
C.8 A three stage switch for $N = 6$, $c = 4$, and $t_A(i) = 2$ for all nodes $i$. . . . 139
C.9 A hierarchical ring with parameter $\alpha = 2$, $\tau = 1$, and $L = 3$. . . . . 140
C.10 The segment of subnet $s$. . . . . 142
C.11 A parent subnet $s$ composed of its two children subnets $s_0$ and $s_1$. . . . . 142
C.12 Relative cost $W/W_{\min}$ for $N = 8$, $c = 16$, and $g = 1, 2, ..., 16$. . . . . 151
C.13 Relative cost $Q/Q_{\min}$ for $N = 8$, $c = 16$, and $g = 1, 2, ..., 16$. . . . . 152
C.14 Relative cost $W/W_{\min}$ for $N = 16$, $c = 16$, and $g = 1, 2, ..., 16$. . . . . 154
C.15 Relative cost $Q/Q_{\min}$ for $N = 16$, $c = 16$, and $g = 1, 2, ..., 16$. . . . . 155

D.1 An optical WDM ring. . . . . 157
D.2 An optical node. . . . . 158
D.3 The segment of subnet $s$. . . . . 160
D.4 A parent subnet $s$ composed of its two children subnets $s_0$ and $s_1$. . . . . 161
D.5 A parent subnet $s$ composed of its two children subnets $s_0$ and $s_1$ and the placement of ADMs. . . . . 162
D.6 For a ring network with $N = 16$ and $W = 32$, the plots give the average values of $tAVG$, $tMIN$, $tMAX$, and $ADMAVG$. . . . . 167
D.7 The ratio $ADMAVG/tAVG$ as a function of $F$ for three different values of $N$. The curves from top to bottom are for $N = 16, 12$, and 8, respectively. 168
D.8 The ratio $ADMAVG/W$ as a function of $F$ for three different values of $N$. The curves from top to bottom are for $N = 8, 12$ and 16, respectively. 169

E.1 An opaque node. . . . . 171
E.2 An optical node. . . . . 172
E.3 NSFNET. . . . . 174
E.4 Table showing the average minimum cost for different ranges of values of $\{t(i)\}$. . . . . 175
E.5 A traffic stream route. . . . . 176
E.6 A star subnetwork with an optical node at the hub. . . . . 177
E.7 The DROP algorithm. . . . . 179
E.8 Sensitivity of DROP with parameter $c$. . . . . 181

# Chapter 1

# Summary

*Wavelength division multiplexing* (WDM) is the practical optical technology that exploits the bandwidth in optical fiber. It allows multiple optical signals, carried at different *wavelengths*, to be multiplexed onto fibers. In this way, a physical fiber can operate as many multiples of *virtual fibers*.

WDM networks provide a means to support very high bandwidth traffic. However, today and in the near future, equipment costs are high. Components that can dominate costs include multiplexers, demultiplexers, and switches. Reduction in equipment can lower cost, but it can also restrict the bandwidth efficiency of the network. Thus, it is desirable to have networks that can achieve or nearly achieve 100% efficiency, while employing limited switching.

This project studied the problem in two layers of WDM networks. These are referred to as the *optical* and *tributary traffic* layers. At the optical layer, optical connections, known as *lightpaths*, are switch between end nodes. These lightpaths are constrained by the kind of switching available at intermediate nodes. One type of switching, known as *wavelength conversion*, allows an optical signal to change its *wavelength*, which is the inverse of the carrier frequency.

We considered networks that allow a limited amount of wavelength conversion. Presumably the limited conversion would be cheaper to implement than a network that allows *full conversion* (i.e., all possible conversions), but may approach the performance of a network with full conversion.

For limited conversion wavelength networks, we investigated whether the pattern of wavelength conversion could influence performance. By simulation we show that networks that shift wavelengths within a local range perform better than networks that spread the wavelength shifts over all other wavelengths.

We also provided resource allocation algorithms that find channels for lightpaths. We proposed and evaluated channel assignment algorithms given pre-computed routes. We proposed and found algorithms that rearranged existing lightpaths to make way for new ones, and algorithms that found channel assignments without pre-computed routes.

At the tributary traffic layer, tributrary electronic data streams are switched between end nodes. These are *constant bit rate* connections that are multiplexed into lightpaths. Thus, the traffic switching is at a finer grain. For example, the tributary traffic streams

could be OC-3 (155 Mbps) while the lightpaths could be OC-48 (2.5 Gbps). Thus, 16 traffic streams can be multiplexed into a lightpath. Notice, at this layer, lightpaths are the physical links of the network.

We investigated networks that allow optical signals to pass-through nodes. This avoids expensive transponders and electronic multiplexing equipment. However, it may reduce the switching capability at the node so that the network becomes bandwidth inefficient. However, we found networks that do not lose bandwidth efficiency, under a certain restriction on the traffic. The restriction is that traffic data streams once set up, never terminate. This is not unrealistic since thse connections are high rate (e.g., OC-3), carrying many individual user connections. If traffic is allowed to terminate then the network is still bandwidth efficient, but may have to rearrange existing traffic to make way for new. The cost savings of this network can be significant. We show in simulations that it can decrease line terminating equipment costs by 44% from networks without optical pass-through.

# Chapter 2

# Introduction

Optical fiber communication is the primary transmission medium for backbone communication in networks. Single-mode optical fiber, supports very high bandwidth (on the order of tens of Terahertz) at very low bit error rates ($10^{-9}$ or less). *Wavelength division multiplexing* (WDM) is the technology that makes it practical to utilize the bandwidth in fiber. WDM is basically *frequency division multiplexing* (FDM) for optical signals. A single fiber can support multiple optical signals, each at different carrier frequencies. At present, the practical transmission rates of each channel are at the OC-48 (2.5 Gbps) or OC-192 (10 Gbps). In the language of WDM technology, these optical channels are said to be carried at different *wavelengths*, which are just the inverse of the carrier frequencies. WDM fiber links are commercially available today at 80 wavelengths, and systems of 100s of wavelengths may be available in the very near future.

A WDM network can be classified into those that do not allow optical signals to pass-through a node, and those that do. We will refer to networks that do not as *point-to-point* networks, and those that allow pass-through as *wavelength routed* networks.

*Point-to-point* networks terminate each WDM channel with electronic terminating equipment. In other words, each optical signal traverses exactly one fiber link. This is the how WDM is used today; it enables a single fiber to operate as multiple *virtual fibers*, corresponding to different wavelengths.

An example of a point-to-point network node is shown in Figure 2.1. The node is composed of WDM *multiplexers* (Muxes), WDM *demultiplexers* (Dmuxes), *line terminating equipment* (LTE), and *digital crossconnect systems* (DCSs). The WDM Muxes/Dmuxes multiplex and demultiplex optical signals at different wavelengths onto and off of fiber links. LTEs and DCSs process electronic tributary data streams, that are transported

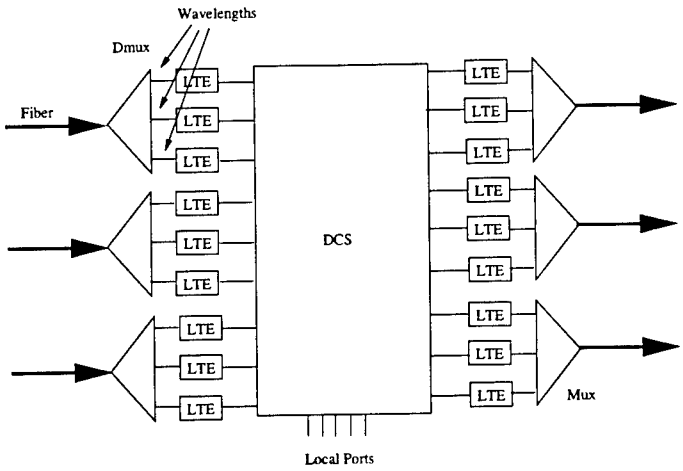



Figure 2.1: A node in a point-to-point network.

over the WDM channels. For example, these electronic tributary streams could be SONET OC-3 (155 Mbps) data streams, while the WDM channels could have a line rate of OC-48 (2.5 Gbps). LTEs multiplex and demultiplex the tributary streams onto and off of the WDM channels, and the DCSs switch the tributary streams between LTEs and local ports.

This architecture has the advantage of maximum switching capability at the nodes. It can switch at the level of the tributary traffic streams, and between any WDM channels. This supports very efficient use of the fiber bandwidth. However, the architecture is costly. LTEs and DCSs can be expensive. This is especially true in systems with large numbers of wavelengths, and when a large fraction of traffic just passes through nodes. Thus, there can be considerable cost savings if the network allows optical signals, that carry transit traffic, to pass through nodes and avoid the LTE and DCS costs.

*Wavelength routed networks* allows optical pass-through. An optical signal can traverse multiple fiber-links. A *lightpath* refers to the optical end-to-end connection that an optical signal follows. Figure 2 shows a wavelength routed network and two lightpaths $L1$ and $L2$. $L1$ traverses three links, all at the same wavelength $w_2$. To support the lightpath, each node has WDM multiplexers and demultiplexers to multiplex and demultiplex the signal onto and off of fibers. Nodes also have optical switches that switch optical signals between the multiplexers and demultiplexers.

$L2$ also goes through three links, but uses two different wavelengths $w_1$ and $w_3$. To support this lightpath, node 3 must have a *wavelength converter*, that shifts the lightpath

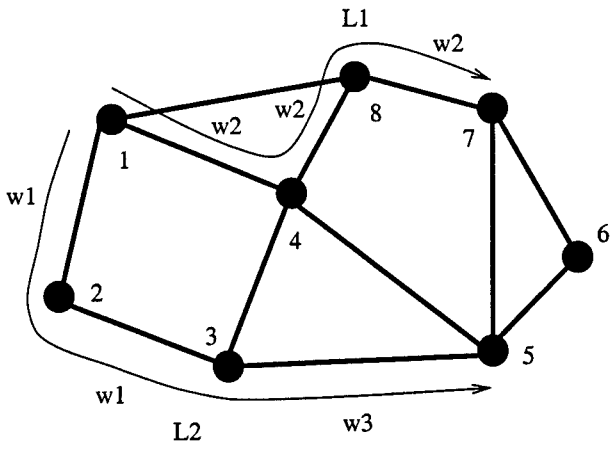



Figure 2.2: A network with two lightpaths $L1$ and $L2$.

from $w_1$ to $w_3$. Note that networks with wavelength converters can more efficiently utilize their channels. For example, consider a lightpath that must traverse two links, but where the first link has a free channel only at wavelength $w_1$, and the second link has a free channel only at wavelength $w_3$. Without a wavelength converter at the intermediate node, the lightpath cannot be supported.

Figure 2.3 shows an example node in a wavelength routed network. Its components include WDM Muxes/Demuxes, *wavelength converting devices (WCDs)*, *optical cross connects (OXCs)*, LTEs, and DCSs. WCDs shift the wavelength of an optical signal. Optical crossconnects take optical signals at their incoming ports and switch them to their outgoing ports. They can be *wavelength selective* or *wavelength converting*. Wavelength selective means that the optical signal keeps its wavelength through the switch. Wavelength converting means that the optical signal can change wavelengths.

Notice that there are two layers of switching in the node which we will refer to as the *optical layer* and *tributary traffic layer*. The optical layer switches lightpaths, and is composed of optical Muxes/Demuxes, OXCs, and wavelength converting devices. It uses the fibers as its physical links. The tributary traffic layer switches the tributary traffic, and is composed of the LTEs and DCSs. It uses the lightpaths as its physical links.

This project investigates efficient designs for wavelength routed networks, at the optical and tributary traffic layers. At the optical layer we studied how wavelength conversion can be used to efficiently support lightpaths. The scope of this part of the project is discussed in Section 2.1. At the tributary traffic layer, we study how LTEs, DCSs, and OXCs can be placed to efficiently support tributary traffic streams. The scope of this

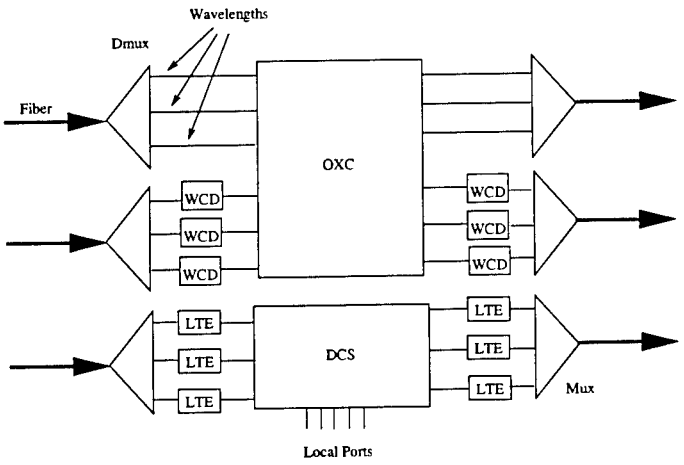



Figure 2.3: A node in a wavelength routed network.

part of the project is discussed in Section 2.2. In both layers we propose network architectures that have a minimal amount of switching (or wavelength shifting). We compare the performance of these architectures with those that have a full complement of switching capabilities. In Section 2.3, we describe the organization of the rest of this report.

## 2.1 Limited Wavelength Conversion Networks

Wavelength converters (also known as *wavelength translaters* or *wavelength shifters*), can either operate using *all-optical* or *elecro-optical-electrical* (E-O-E) technology. All-optical devices keep the optical-signal optical as its being shifted. E-O-E devices convert the optical signal (which is an analog signal) into an electrical signal, and then converts the electrical signal back into an optical signal. While these conversions are done, the signal is preserved as an analog signal, i.e., it is not converted into a digital bit stream at any point. The E-O-E technology has the advantage of being practical to implement today, though it has the disadvantage of being bandwidth limited by electronic technology.

Network nodes are classified into three types: *no*, *full*, and *limited* conversion as shown in Figure 2.4. No conversion means that the node has no wavelength converters, and is the simplest to implement. Full conversion means that the node can convert any of its incoming optical signals to any wavelength. This node provides the maximum amount of wavelength converting capability and bandwidth efficiency, but at a cost of more implementation complexity. Limited conversion is between the two extremes. It can provide some conversion capability, but at less complexity than full conversion.

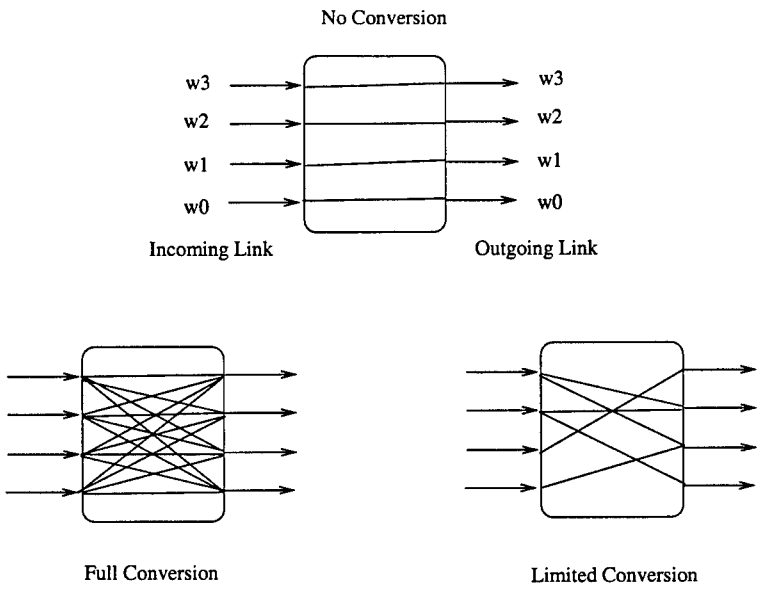



Figure 2.4: Examples of wavelength conversion.

It was shown in [26] that for the ring topology, a network with limited conversion at one node and no conversion at the other nodes can perform as well as a network with full conversion at every node. The limited conversion node allowed a wavelength to be shifted by two wavelengths (up or down). For example, an incoming signal with wavelength $w_k$ can be shifted to one of the wavelengths $\{w_{k-2}, w_{k-1}, w_k, w_{k+1}, w_{k+2}\}$. Although the results showed that limited conversion can be beneficial, it assumed lightpaths may be *rearranged.* Thus, if a new lightpath is to be set up then other existing lightpaths may have to be rearranged to make room for it. This limits the applicability of the results because in practice, connections are never rearranged to avoid disruption of service.

This part of the project is an architectural study of wavelength routed networks with limited conversion. It investigates how efficiently these networks perform for dynamic traffic when lightpaths cannot be disturbed, or if they can be rearranged, then only to a limited extent. The bandwidth efficiency of these limited conversion networks will be compared with full conversion networks. It investigates a couple of important aspects of a network:

**Channel Attachment Patterns (CAP):** This defines how an optical signal may be wavelength converted at a node. Figure 2.4 shows CAPs for three types of nodes: no, full and limited wavelength conversion. For example, in the figure, the case of limited wavelength conversion allows an incoming signal at wavelength $w_3$ to be

shifted to wavelengths $w_1$ and $w_2$, but not to any other wavelengths. A measure of the complexity of a CAP is its *wavelength degree* (or *w-degree*), which is defined to be the maximum number of wavelengths an incoming signal may be shifted to. For example, in Figure 2.4, the limited conversion node has w-degree 2. Also note that a node with full conversion has w-degree equal to the total number of wavelengths, and a node with no conversion has w-degree 1.

There are two types of CAPs that we considered: *local* and *expanding*. Local CAPs allow wavelength shifts that are restricted in range. For example, an incoming signal at a wavelength $w_k$ (for some $k$) may be allowed to only shift to wavelengths $\{w_{k-1}, w_k, w_{k+1}\}$. Expanding CAPs do not have the limited range restriction, and tend to spread signals to other wavelengths in a uniform way.

The specific CAPs we studied are discussed in Subsection 2.1.1. Included in the subsection is a discussion of an optimization algorithm that designs CAPs.

**Channel Assignment Algorithms:** To set up a lightpath in a wavelength routed network, a sequence of WDM channels must be found for it. Each consecutive pair of channels in the sequence are constrained according to the wavelength conversion capabilities at their common node. In particular, the wavelength of an incoming channel must be wavelength convertible to the wavelength of the outgoing channel. We refer to such a pair of channels as being *w-attached*. We refer to a sequence of channels for a lightpath as a *channel assignment* for the lightpath.

We examine three types of channel assignment algorithms.

- *Channel assignment for pre-computed routes and no rearrangement.* This type of channel assignment assumes that a lightpath request comes with a pre-computed route. This route may have been computed by another algorithm, e.g., a shortest hop routing algorithm. The task of the algorithm is to find the wavelengths to be used by the lightpath along the route.

  In addition, the channel assignment will not disturb existing lightpaths. In other words, the assignment must be of free channels not used by existing lightpaths. Subsection 2.1.2 discusses these algorithms in more detail.

- *Channel assignment for pre-computed routes and with rearrangement.* This

type of channel assignment also assumes that a lightpath request comes with a pre-computed route. However, it is allowed to rearrange the channel assignments of existing lightpaths to make way for a new one. The rearrangements are designed to minimize disruption of existing lightpaths.

We were motivated to investigate these algorithms since they should improve the network performance over algorithms that do not disturb existing lightpaths. In addition, it is expected that a large fraction of the capacity of optical networks will carry Internet traffic, which is highly tolerant to link and node distruptions of short duration. Thus, disturbing existing lightpaths may be tolerable for Internet traffic if they are small. Subsection 2.1.3 discusses these algorithms in more detail.

- *Channel assignment given no pre-computed routes.* This type of channel assignment is only given the source and destination nodes of the lightpath. It must compute the route as well as the wavelengths along the route. In addition, it does not disturb existing lightpaths.

  We investigate two ways to do this. The first is to find the route and wavelengths simultaneously. The second is to first find a route, and then given the route, find a channel assignment using one of the above channel algorithms for pre-computed routes. Subsection 2.1.4 discusses these algorithms in more detail.

### 2.1.1 Channel Attachment Patterns (CAPs)

In this section, we describe our study of efficient CAPs. As mentioned earlier, we considered two types of CAPs: local and expanding. These CAPs are described next. Each is parameterized by its w-degree, which is denoted by $d$ in this subsection. We will let $W$ denote the number of wavelengths. The three local CAPs are:

**PARTITION:** The wavelengths are partitioned into subsets of size $d$, except for possibly one, which may be smaller. In particular, the first $d\lfloor \frac{W}{d} \rfloor$ wavelengths are partitioned into $\lfloor \frac{W}{d} \rfloor$ subsets $\{w_0, w_1, ..., w_{d-1}\}, \{w_d, w_{d+1}, ..., w_{2d-1}\}$, and so on, each of size exactly $d$ (recall that $\lfloor \frac{W}{d} \rfloor$ is the largest integer that is at most $\frac{W}{d}$). The remaining wavelengths form a final subset. Wavelength conversion is possible within and only

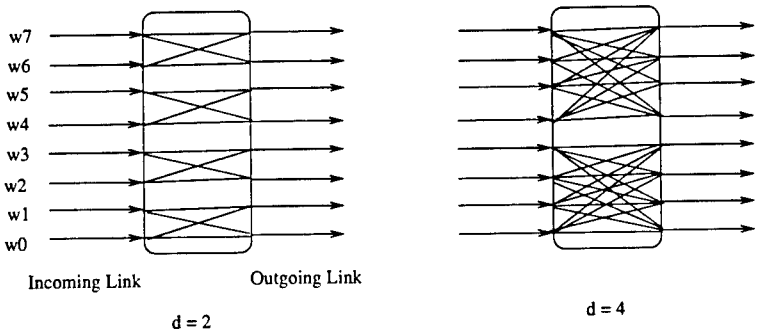



Figure 2.5: PARTITION for $d = 2$ and 4, and $W = 8$.

within a subset. Figure 2.5 shows PARTITION for $d = 2$ and 4.

PARTITION can model wavelength conversion when it is implemented by O-E-O, and the analog electrical signals are switched between wavelengths by an electrical switch. Each subset of wavelengths corresponds to channels that have their optical signals converted to electrical ones, then are switched by a $d \times d$ electrical switch, and then converted back into optical signals. For example, in Figure 2.5 for $d = 4$, wavelengths $\{w_4, w_5, w_6, w_7\}$ have their signals converted to electrical ones, and switched by a $4 \times 4$ switch (e.g., a crossbar), and converted back to optical signals.

Notice that PARTITION does not allow lightpaths to use channels from different subsets, but allows full conversion within subsets. In this way, the network is basically $\lceil W/d \rceil$ parallel full conversion networks, each with $d$ wavelengths, except for perhaps one. (Recall that $\lceil W/d \rceil$ is the smallest integer that is at least $W/d$.)

**SHIFTED:** This is a modification of PARTITION, where some nodes have the same CAP as PARTITION, while others have a "shifted" version. The shifted version, which we refer to as the S-PARTITION, has subsets defined by a shift parameter $s = \lceil \frac{d}{2} \rceil$. For the S-PARTITION, the first subset of wavelengths has size $s$, the next subsets have size exactly $d$, except possibly the last one. Just as in PARTITION, wavelength conversion in S-PARTITION is possible within and only within subsets. Figure 2.6 shows the CAPs for SHIFTED when $d = 2$ and 4.

The possible advantage of SHIFTED over PARTITION is that it allows lightpaths to cross the wavelength partition boundaries of PARTITION.

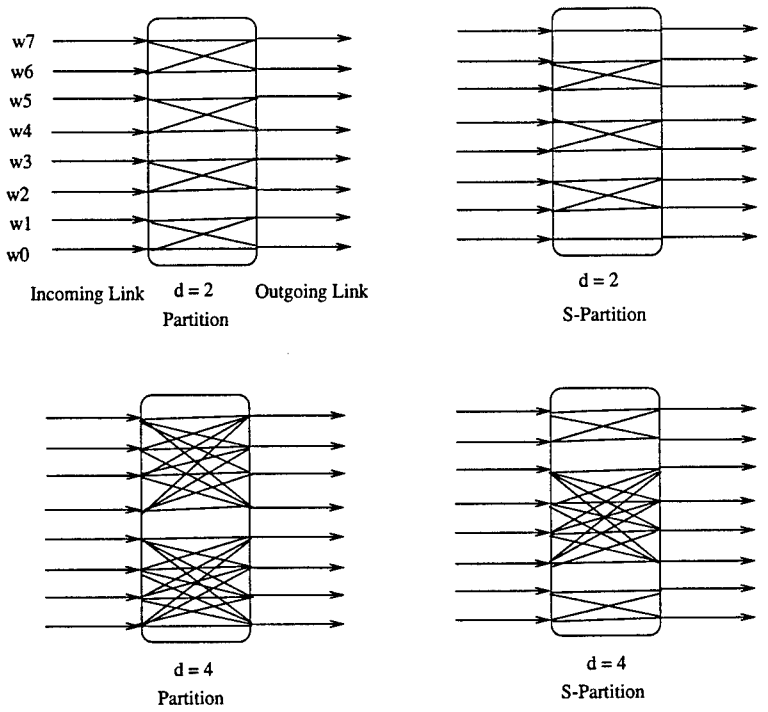



Figure 2.6: SHIFTED for $d = 2$ and 4, and $W = 8$.

**DISTRIBUTE:** Each optical signal may be converted to an optical signal within a limited range. In particular, if the optical signal has wavelength $w_k$ (for some $k$) then it can be shifted to a wavelength $w_j$ if and only if $k - \frac{d}{2} < j \leq k + \frac{d}{2}$, as shown in Figure 2.7. DISTRIBUTE models the all-optical wavelength conversion technology of *four-wave mixing* [34]. This is sometimes referred to as *limited range* conversion [34].

The next set of CAPs are the expanding ones, where optical signals can be can be shifted over a wide range.

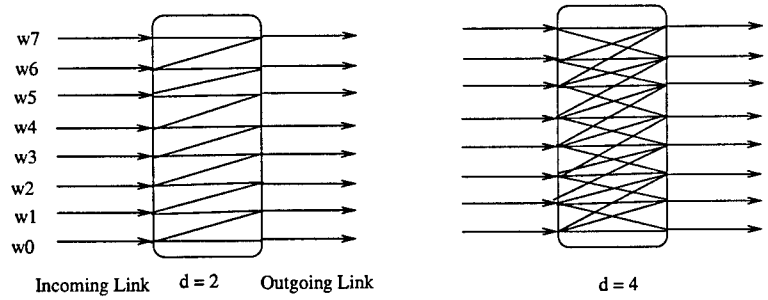



Figure 2.7: DISTRIBUTE for $d = 2$ and 4, and $W = 8$.

**RANDOM-SAME:** This is randomly chosen, where each incoming channel can be shifted to exactly $d$ outgoing channels, and each outgoing channel can be shifted from exactly $d$ incoming channels. The CAP is randomly chosen among all possible, where each is equally likely. Note that it is assumed that the same CAP is used at all nodes in the network.

One way to implement this to have each incoming WDM channel be terminated by a receiver that converts the signal to an electrical one. The receiver is connected to a demultiplexer that has one input and $d \cdot (D-1)$ output ports, where $D$ is the number of outgoing (and incoming) fibers. Also each outgoing WDM channel has an electrical multiplexer with $d \cdot (D-1)$ input ports and one output. The output is connected to a transmitter which converts the electrical signal to an optical one for the WDM channel. Then the multiplexers and demultiplexers are connected to form the CAP. Thus, each demultiplexer (resp., multiplexer) of an incoming (resp., outgoing) fiber is connected to $d$ multiplexers (resp., demultiplexers) of an outgoing (resp., incoming) fiber.

**RANDOM-RANDOM:** This is similar to RANDOM-SAME, but each node has a (possibly) different CAP chosen for it that is statistically independent of the other nodes.

**SHUFFLE:** A signal at wavelength $w_k$ (for all $k$) may be shifted to the wavelengths

$$\{w_{(d \cdot k+j) \bmod W} : j = 0, 1, ..., d-1\}$$

as shown in Figure 2.8 (recall $(d \cdot k + j) \bmod W$ is the remainder of $d \cdot k + j$ divided by $W$). This type of pattern has been used in interconnection networks for parallel processing. It allows a signal at a wavelength to convert to a large number of wavelengths after traversing only a few links.

The expanding CAPs allow a signal at one wavelength to reach a larger number of channels after traversing a few links compared to the local CAPs. It was hoped that this would make it easier to find free channels when setting up a lightpath.

We also considered an algorithm that designs CAPs which we refer to as algorithm *DesignCAP*. The purpose of this part of the study was to see what kind of "optimal" designs would come of it.

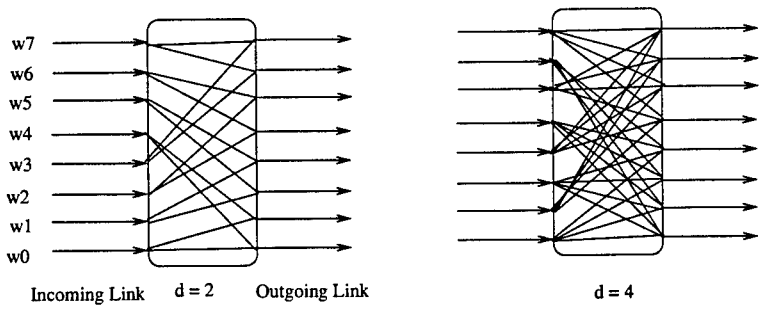



Figure 2.8: SHUFFLE for $d = 2$ and 4, and $W = 8$.

DesignCAP is a simple *local search* algorithm. It searches through a collection of CAPs, where the CAPs are defined by a particular number of wavelengths $W$ and w-degree $d$. In particular, the collection is the set $C$ of all CAPs where each incoming wavelength is w-attached to exactly $d$ outgoing wavelengths, and each outgoing wavelength is w-attached to exactly $d$ incoming wavelengths. Each of these CAPs $c \in C$, has a *cost* value $f(c)$. This cost reflects the performance of a network given that each node uses $c$. The algorithm attempts to solve the problem $\min_{c \in C} f(c)$, by searching through the space of $C$.

The search starts from an arbitrary CAP. Then it goes through a sequence of CAPs, each one being a "perturbation" of the previous one. Figure 2.9 shows what a "perturbation" is. On the left you have a CAP with its pairs of wavelengths that are w-attached. A *branch exchange* is where you take two pairs of w-attached wavelengths such that the pairs have no common incoming wavelength or common outgoing wavelength, and have the pairs swap wavelengths. This leaves a CAP with the same w-degree. An example branch exchange is shown in the figure. A branch exchange is a "perturbation."

**Description of DesignCAP:**

The algorithm is an iterative algorithm as shown in Figure 2.10. At each iteration, the algorithm has a "current" CAP $c \in C$. All perturbations $c'$ of $c$ are considered, and their costs $f(c')$ computed. If all of the perturbations have cost at least as high as $f(c)$ then the search stops and $c$ is the solution returned by the algorithm. (This means $c$ is a local minimum solution.) Otherwise, the perturbation with smallest cost becomes the new $c$, and the search continues through another iteration. □

DesignCAP intelligently searches through CAPs to find ones that have low cost. Its time complexity can be much lower than exhaustively considering all possible CAPs.

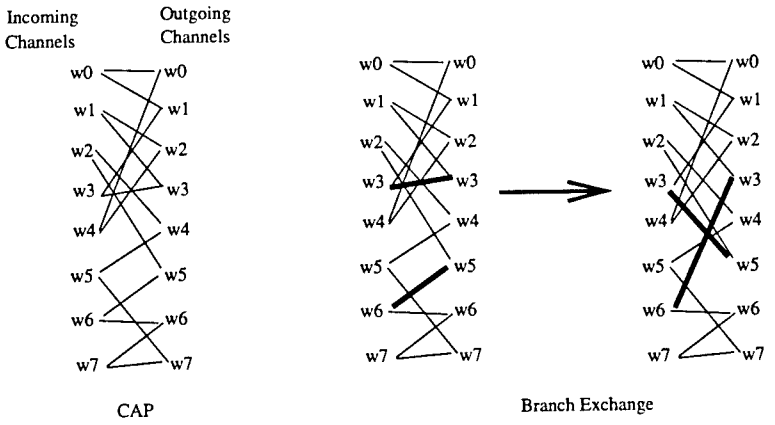



Figure 2.9: A CAP and one of its branch exchanges.

**DesignCAP**

1. Let $c$ be an initial (arbitrary) CAP.
2. Find the perturbation (i.e., branch exchange) $c'$ of $c$ that minimizes $f(c')$. Let $c_{\min}$ denote this perturbation.
3. If $f(c_{\min}) \geq f(c)$ then return $c$ and STOP.
4. Otherwise, let $c = c_{\min}$, and go to Step 2.

Figure 2.10: Algorithm DesignCAP.

The number of all possible CAPs grows exponentially with $W$. On the other hand, each iteration of DesignCAP considers at most $Wd \cdot (Wd-1)/2$ branch exchanges.

However, DesignCAP still requires high processing time because we have a brute force method of evaluating cost $f$. We find $f(c)$ by running a simulation of a small WDM unidirectional ring network which has $c$ for the CAP at each node. Then $f(c)$ is the measured blocking probability. We chose to simulate a ring because even a small sized network (say of 8 nodes) can have long lightpaths (more than three or more hops). Such a network with long lightpaths is desirable because short lightpaths do not require much wavelength conversion.

### 2.1.2 Channel Assignment for Pre-Computed Routes and No Rearrangement

In this subsection we describe channel assignment algorithms for lightpaths that have pre-computed routes. Thus, when a lightpath is set-up, the network is responsible only for determining the wavelengths along the route. In addition, the set up cannot disturb existing lightpaths. Note that a consecutive pair of channels in the assignment must conform to the CAP at their intermediate node, i.e., they must be w-attached. Figure 2.11(a) shows a route (which is the sequence of links Link 1, .., Link 4) and the channels along the route.

Note that the problem of finding a channel assignment in Figure 2.11(a) is the problem of finding a "path" of free channels from the source and to the destination node. The problem can be converted to the problem of finding a path in a *directed graph* that represents the channels. We refer to this graph as the *channel-graph*. In the channel-graph, there is an edge for each channel which is referred as a *wavelength edge*. Each wavelength edge is incident to two vertices, which represents the ports of the channel. Wavelength edges are connected by other edges, referred to as *mapping edges*. These mapping edges indicate how channels are w-attached. For example if an incoming channel that is w-attached to an outgoing channel at a node then their corresponding wavelength edges are connected by a mapping edge. There are also *end edges*, which connect the source to each of its outgoing channels, and connect the destination to each of its incoming channels. Figure 2.11(b) shows the corresponding channel-graph for Figure 2.11(a). Notice that the number of vertices in the channel-graph is $2+2C$, where $C$ is the number of channels.

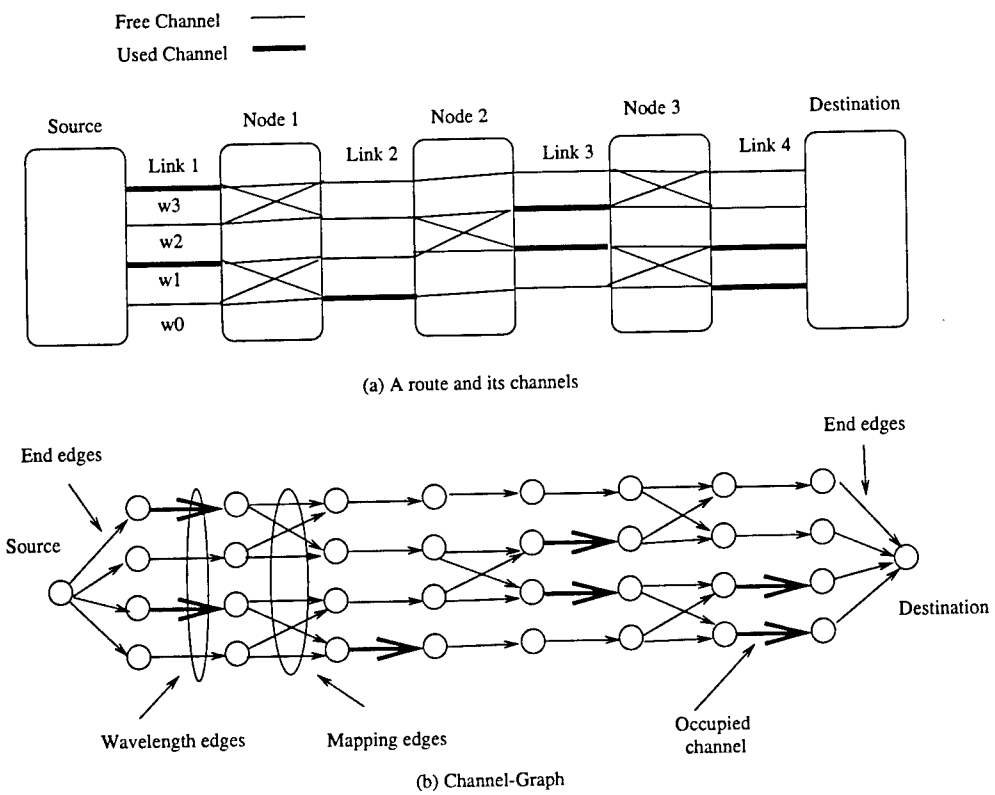



Figure 2.11: Channels along a lightpath's route.

To find a route of free channels in a channel-graph, standard graph algorithms can be used to find paths. We employ two of these, *depth first search* (DFS) [32] and *a shortest path algorithm* [32], to realize the channel assignment algorithms.

**Depth First Search (DFS)**

Depth first search (DFS) starts at the source and searches forward, always exploring free and unexplored channels. If it gets stuck along the way, it *backtracks* (or *cranks back*) until it can go forward again. For example, in Figure 2.11, and depth first search may occur as follows. It starts from Link 1 and $w0$, proceeds to Link 2 at $w_1$. However, then it gets stuck because there are no free channels on Link 3 to proceed to. The search backtracks all the way to the start node. Then it tries $w_2$ on Link 1, and proceeds to $w_2$ on Link 2. Again it gets stuck due to no free channels on Link 3. It backtracks, but only to Link 1. Now it tries $w_3$ at Link 2. Then it can proceed all the way to the end node by following the channels at $w_3$ on Links 3 and 4.

Note that depth first search has some flexibility in the way it can be implemented. As it is searching forward, it may have more than one free channel it can proceed along.

If there are multiple choices of channels, our DFS algorithm gives preference according to wavelengths. We consider four types of preferences, but all have the same structure. Each wavelength $w_i$ is given a preference value $p_i$. A wavelength's preference improves with smaller values of $p_i$. Channels with the more preferred wavelengths are chosen first in DFS. The following are the four wavelength preferences we considered.

**Fixed:** For each wavelength $w_i$, its preference value is $p_i = i$. This preference is also known as *first-fit.*

**Random:** Before each lightpath set-up, the preferences of the wavelengths are randomly chosen where each ordering is equally likely.

**Pack:** Before each lightpath set-up, the *utilization* of each channel is determined, where the utilization is the fraction of time that the channel was busy. Then the utilization $u_i$ of each wavelength $w_i$ is computed which is defined to be the average of the utilizations over all channels at the wavelength. Let $u_{max}$ be the maximum utilization over all wavelengths, i.e., $\max_i u_i$. Then the preference $p_i$ for wavelength $w_i$ is $1 - \frac{u_i}{u_{max}}$. Notice that if the wavelength has high utilization then it has high preference.

**Spread:** This is just the opposite of Pack. It computes the same utilization value $u_i$ for each wavelength $w_i$, but gives preference to those wavelengths with low utilization.

The Fixed and Pack preferences tend to keep lightpaths to a small set of wavelengths, while Random and Spread tend to spread them out. In [22], these channel assignment preferences were studied on networks with no conversion. It was shown that Fixed and Pack performed performed about the same or better than Random and Spread. (Actually in [22], Spread and Pack are implemented differently. They measured utilization at the current time, while we measure utilization averaged from the initial time.)

The time complexity of the classical depth first search is $O(E)$, where $E$ is the number of edges. For a channel-graph, $E$ is $O(LW^2)$, where $L$ is the length of the route. Thus the DFS algorithm has time complexity $O(LW^2)$ to do the search. This does not include the time to compute the wavelength preferences. Determining wavelength preferences will require sorting the values $\{p_i\}$. This takes $O(W \log W)$. For Fixed, Pack, and Spread, this is sufficient processing for the wavelength preferences. For Random, a

random permutation has to be computed too. This can be done in $O(W^2)$, assuming a (pseudo) random integer can be computed in $O(1)$ time. Then determining the wavelength preferences takes $O(W^2)$ time. Thus, the DFS channel assignment takes $O(LW^2)$ time. Note that $L < N$ unless the route has loops, where $N$ is the number of nodes in the physical network. Then the time complexity is $O(N^2W^2)$.

**Shortest Path Algorithm (SP)**

For this channel assignment, each free channel is assigned a *cost*. The channel assignment found for the lightpath is the one with the minimum total cost. The cost of a free channel is a function of the utilization $u_i$, where $w_i$ is the wavelength of the channel. In particular, the weight is $(1 - u_i/u_{max})^a$, where $a$ is some power to be chosen to optimize performance. If $a = 1$ then the algorithm finds a channel assignment which has the maximum sum of wavelength utilizations. If $a$ is very large then the algorithm finds a channel assignment that includes the most number of channels with high utilization. The cost of an occupied channel is infinity.

To find a channel assignment, a channel-graph can be constructed where the vertices (representing free channels) are given costs. A shortest path algorithm, and in particular Dijkstra's algorithm [8], is used to compute a minimum cost path. Since the channel graph has $O(L \cdot W)$ vertices, where $L$ is the length of the route, the time complexity of Dijkstra's algorithm to find the minimum cost path is $O(L^2 \cdot W^2)$. Thus, the time complexity to find the channel assignment is $O(L^2 \cdot W^2)$. Note that $L < N$ unless the route has loops, where $N$ is the number of nodes in the physical network. Then the time complexity is $O(N^2W^2)$.

### 2.1.3 Channel Assignment With Pre-Computed Routes and Rearrangement

In this subsection, we will discuss the channel assignment algorithms that assume lightpaths have pre-computed routes. In addition, existing lightpaths can have their channels rearranged to make way for new ones. However, it is assumed that existing lightpaths that are rearranged cannot change their routes. The motivation for this last constraint is that the route may have been chosen for a specific property. For example, it may have been chosen to minimize lightpath congestion on fiber links. Another example, is the lightpath may carry protection bandwidth in case of a fiber fault. Then the lightpath

may have its path chosen to avoid the fibers it is protecting.

A lightpath may have its channels rearranged in one of two ways [21].

**Rearrangement after shutdown (RAS) rearrangement.** The lightpath is terminated (or shut down), freeing up its existing channels. Then it is set up again using its new channels. Note that RAS means the lightpath has a disruption of service.

**Move-to-vacant (MTV) rearrangement.** The lightpath's new channels are free (or vacant). The lightpath can now use its new channels to carry its signal. Then the lightpath can terminate and free up its old channels. Note that MTV means that a lightpath can change channels with minimal disruption of service.

For MTV the new channels of a lightpath are disjoint from the old channels. In the case of RAS, the new and old channels may have common elements.

We consider three channel assignment algorithms, which are referred to as the *RAS*, *Single-MTV*, and *Multiple-MTV* algorithms. They rearrange channels but by keeping distruption to a minimum. All three try to find a channel assignment for a lightpath request $R$, by first trying to find a channel assignment of only free channels. This is done using the DFS algorithm with Fixed wavelength priorities. From now on we will refer to this as DFS/Fixed, since it will be referred to often.

If DFS/Fixed fails then they try to find one or more existing lightpaths to rearrange to make way for $R$. In the case of RAS and Single-MTV, only one existing lightpath is rearranged. In the case of Multiple-MTV, multiple existing lightpaths may be rearranged. In the case of the RAS algorithm, the existing lightpath is rearranged using RAS rearrangement. In the case of Single- and Multiple-MTV, the existing lightpaths are rearranged using MTV rearrangement.

**RAS Algorithm** All existing lightpaths that share a common fiber link with the route of $R$ are referred to as *common lightpaths*. Note that these are the lightpaths whose rearrangement could lead to a channel assignment for $R$. Let $P_1, P_2, ...P_k$ be the collection of common lightpaths with $R$, where $k$ is the number of such existing lightpaths.

Each lightpath $P_i$ is considered for rearrangement until one is found that will lead to a channel assignment for $R$. In particular, a lightpath $P_i$ is considered by first freeing

**RAS Algorithm**
Input: A lightpath request with route $R$.
Output: A channel assignment for $R$, and a new channel assignment for any lightpath that should be rearranged.

1. **if** a channel assignment for $R$ using DFS/Fixed can be found **then** return it and exit.
2. Find the common lightpaths $P_1, P_2, ..., P_k$ of $R$, where $k$ is the number of such common lightpaths.
3. **for** $i = 1$ **to** $k$ **begin**
   - Mark all channels of $P_i$ as *pseudo free.*
   - Find a channel assignment $P'$ for $R$ (using DFS/Fixed) that is composed only of free and pseudo free channels.
   - **if** such a channel assignment can be found **then begin**
     - Find a new channel assignment for $P_i$ (using DFS/Fixed) that is composed of free or pseudo free channels but is disjoint from the channels of $P'$.
     - **if** such a channel assignment can be found **then** return the channel assignment for $R$ and the new channel assignment for $P_i$. Then exit.
     - **end**
   - Unmark the channels of $P_i$, so they are not pseudo free.
   - **end**
4. Return "algorithm is unsuccessful". exit.

Figure 2.12: Rearrangement-after-shutdown (RAS) algorithm.

its channels. Then a channel assignment of free channels is found for $R$. Subsequently a channel assignment is found for $P_i$ from the remaining free channels. If both channel assignments can be found then they are returned.

Pseudo code for the algorithm is given in Figure 2.12.

**Single-MTV** This is a variation of the RAS algorithm. We are given a lightpath request with route $R$. Again the common lightpaths $P_1, P_2, ..., P_k$ are identified. For each $P_i$, we try to find a channel assignment for $R$ assuming the channels of $P_i$ are available. Then we look for a new channel assignment for $P_i$ using only free channels (avoiding the current channels of $P_i$) and excluding the channels for $R$. If both channel assignments can be found then we can stop.

The pseudo code for the algorithm is given in Figure 2.13.

**Single-MTV Algorithm**
Input: A lightpath request with route $R$.
Output: A channel assignment for $R$, and a new channel assignment for any lightpath that should be rearranged.

1. **if** a channel assignment for $R$ using DFS/Fixed can be found **then** return it and exit.
2. Find the common lightpaths $P_1, P_2, ..., P_k$ of $R$, where $k$ is the number of such common lightpaths.
3. **for** $i = 1$ **to** $k$ **begin**
   - Mark the channels of $P_i$ *pseudo free.*
   - Find a channel assignment $P'$ for $R$ (using DFS/Fixed) that is composed only of free and pseudo free channels.
   - **if** such a channel assignment can be found **then begin**
     - Find a new channel assignment for $P_i$ (using DFS/Fixed) that is composed of free channels and is disjoint from the channels of $P'$.
     - **if** such a channel assignment can be found **then** return the channel assignment for $R$ and the new channel assignment for $P_i$. Then exit.
     - **end**
   - **end**
   - Unmark the channels of $P_i$, so they are not pseudo free.
4. Return "algorithm is unsuccessful". exit.

Figure 2.13: Single move-to-vacant (MTV) algorithm.

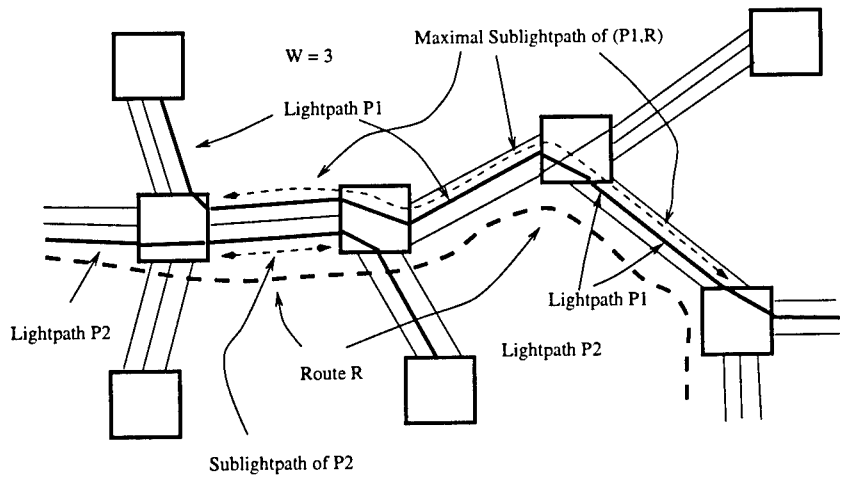



Figure 2.14: Concept of sublightpath and maximal sublightpath.

**Multiple MTV** Multiple-MTV may rearrange more than one lightpath, in MTV fashion, to make way for a new one. The algorithm first tries to find a channel assignment for a lightpath request using DFS/Fixed. If that fails then it determines a channel assignment for the lightpath where the channels may also be occupied. The existing lightpaths corresponding to the occupied channels are considered in some order. When a lightpath is considered, a new channel assignment is computed for it which is disjoint from its old set of channels. If this can be done for all these lightpaths then the algorithm is successful. Then it has a channel assignment for the new lightpath request and it has new channel assignments for all lightpaths, that must be rearranged to make way for the request. In addition, these new assignments will facilitate MTV rearrangement. That was a brief description of the Multiple-MTV algorithm. The algorithm is a modification of the one introduced in [21], which was designed for no conversion networks only.

We will give a more detailed description of the algorithm after some definitions. Consider a lightpath $P$. A *sublightpath* of $P$ is a contiguous subset of $P$. Now consider a route $R$ which may not necessarily be the route of $P$. A *sublightpath of* $(P, R)$ is a sublightpath of $P$ that lies entirely in $R$. Thus, it lies in the intersection of $P$ and $R$. A *maximal sublightpath of* $(P, R)$ is a sublightpath of $(P, R)$ that is not contained in any other sublightpath of $(P, R)$. These concepts are illustrated in Figure 2.14.

An existing lightpath is said to be *MTV-rearrangeable* if there is another channel assignment of free channels that follows its route but is disjoint from its current channels. In other words, the lightpath has an alternate set of free channels that it can change to

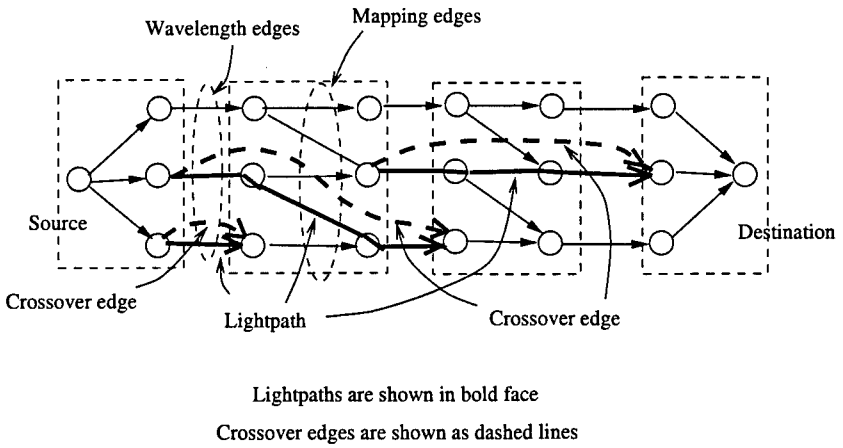



Figure 2.15: MTV sublightpaths and their corresponding crossover edges.

in MTV fashion.

Multiple-MTV finds a channel assignment for a lightpath request with route $R$ in the following way. It identifies all the common lightpaths of $R$. Of these, it identifies the ones that are MTV-rearrangeable. We will refer to these as *common MTV* lightpaths. Then it finds the maximal sublightpaths of the common MTV lightpaths with respect to $R$. We will refer to these as the *MTV sublightpaths.*

Next a channel-graph is constructed along the route $R$. Recall that the graph has wavelength edges (that represent channels), mapping edges (that represent how the channels are w-attached), and end edges (that connect the source and destination to the rest of the channel-graph). New edges are added to the channel-graph referred to as *crossover edges.* The crossover edges represent MTV sublightpaths, so there is one per MTV sublightpath. A crossover edge starts from the beginning of the first channel and ends at the end of the last channel of its corresponding MTV sublightpath. Figure 2.15 shows example crossover edges.

Each of the edges of the channel-graph is assigned a *distance.*

- A mapping or end edge is assigned a distance of zero since there is no penalty for optical switching and wavelength conversion at a node.
- A crossover edge is assigned a distance of one, as a reference distance.
- A wavelength edge that corresponds to an occupied channel with wavelength $w_i$ is given a distance of $\infty$ because it cannot be used.

- A wavelength edge that corresponds to an unoccupied channel is given a distance of $\frac{i}{2WE}$, where $E$ is the number of directed links. This is a small distance to make it preferable over a crossover edge. It is also makes free channels with smaller wavelengths preferable for the channel assignment.

The Multiple MTV algorithm computes a shortest distance path through the channel-graph. If a path cannot be found then the algorithm is unsuccessful. However, if a path is found then its channels are recorded (including the channels in the crossover edges). Let $P$ denote these recorded channels. Let $P_1, P_1, ..., P_k$ denote the lightpaths corresponding to the crossover edges in the path, where $k$ is the number of such lightpaths. The lightpaths are considered for rearrangement according to some order, which is arbitrary. Each of these lightpaths are then MTV rearranged in turn while avoiding the channels of $P$. Note that when a lightpath is MTV rearranged, its old channels are freed for the subsequent lightpaths to use. If they can all be MTV rearranged then the lightpath for $P$ is set up. The pseudo code of the algorithm is given in Figure 2.16.

### 2.1.4 Channel Assignment Without Pre-Computed Routes

The channel assignment algorithms considered in this subsection are ones that are only given the source and destination nodes of a lightpath. We consider three algorithms referred to as *Widest Shortest*, *Shortest Distance*, and *Brute Force Search*.

**Widest Shortest** We will describe the Widest Shortest algorithm after some definitions. A fiber link has a *residual bandwidth*, which is defined to be the number of its free channels. A path has a *residual bandwidth*, which is defined to be the minimum of the residual bandwidths of its links. A *widest shortest* is a shortest hop path over all fiber links that have at least one free channel (i.e., nonzero residual bandwidth). If there is more than one such shortest hop path, the *widest shortest* route is the one with the largest residual bandwidth. The Widest Shortest algorithm is in two steps.

- *Step 1.* The algorithm finds a *widest shortest* route.
- *Step 2.* A channel assignment is found along the route using one of the algorithms in Subsection 2.1.2.

**Multiple-MTV Algorithm.**
Input: A lightpath request with route $R$.
Output: A channel assignment for $R$, and a new channel assignment for any lightpath that should be rearranged.

1. **if** a channel assignment for $R$ using DFS/Fixed can be found **then** return it and exit.
2. Find the common lightpaths $P_1, P_2, ..., P_k$ of $R$, where $k$ is the number of such common lightpaths.
3. Find the MTV sublightpaths.
4. Create the channel-graph with crossover edges along $R$.
5. Find a shortest cost path through the channel-graph. Let $P$ denote these channels.
6. Let $P'_1, P'_2, ..., P'_m$ be the lightpaths corresponding to the crossover edges of $P$, where $m$ is the number of such lightpaths.
7. **for** $i = 1$ **to** $m$ **begin**
   - Compute, using DFS/Fixed, a channel assignment for $P'_i$ that uses free channels and avoids the channels of $P$.
   - **if** unsuccessful **then begin**
     - Restore all the channel assignments of $P'_1, P'_2, ..., P'_i$.
     - Return "algorithm is unsuccessful". Exit.
     - **end**
   - **else begin**
     - Let the lightpath of $P'_i$ be the the new channel assignment.
     - Vacate the old channels of the lightpath of $P'_i$.
     - **end**
   - **end**
8. Now the lightpath for $P$ can be set up.
9. Return the channel assignments $P, P'_1, P'_2, ..., P'_m$. Exit.

Figure 2.16: Multiple move-to-vacant (MTV) algorithm.

Our motivation for considering Widest Shortest routing is that it has been proposed for constrained routing for guaranteed flows of traffic in the Internet [18]. However, a weakness of this algorithm is that during Step 1, the route computation does not take into account the CAP at the intermediate nodes. Therefore, a route may have sufficient free bandwidth on each link, but the corresponding channels may not be w-attached at intermediate nodes.

The time complexity of this algorithm is the sum of the time complexities of the two steps. The first step is to find a widest shortest path. The Bellman-Ford algorithm [8] can be modified to find such a route. Since the Bellman-Ford algorithm has time complexity $O(N^3)$, the first step has time complexity $O(N^3)$. The time complexity of the second step is the time complexity of the channel assignment algorithm of Subsection 2.1.2.

**Shortest Distance** This is also a two step algorithm.

- *Step 1.* A route for the lightpath is computed as follows. Each free channel is given a *distance* value equal to $1/r_e$, where $r_e$ is the residual bandwidth of its fiber link $e$. The algorithm finds a channel assignment of free channels with minimum total distance. We are only interested in the route of this channel assignment.

- *Step 2.* A channel assignment is found along the route using one of the algorithms in Subsection 2.1.2. This channel assignment is returned by the Shortest Distance algorithm.

Notice that the first channel assignment computed (i.e., the one computed in Step 1) is not used because its wavelengths are not optimized. The second channel assignment optimizes the wavelengths. We should also note that the first step is realized by applying Dijkstra's shortest distance algorithm [8] to the channel-graph.

This Shortest Distance algorithm has an advantage over the Widest Shortest algorithm because it does take into account the CAP at intermediate nodes. Thus, the route found in the first step will have a valid channel assignment along it.

The time complexity of the Shortest Distance algorithm is dominated by the time complexity of Dijkstra's shortest path algorithm on the channel-graph. The channel-graph has $O(EW)$ vertices, where $E$ is the number of fiber links and $W$ is the number

of wavelengths. Thus, Dijkstra's shortest path algorithm has time complexity $O(E^2W^2)$. The time complexity for the Shortest Distance algorithm is therefore $O(E^2W^2)$.

**Brute Force Search** This is a modification of the classical *breadth first search* algorithm [8]. It also prioritizes channels according to wavelengths as in the DFS algorithm. The wavelength preferences could be Fixed, Pack, Spread, or Random.

Brute Force Search selects a channel assignment that is the shortest possible (in number of hops). Since it is likely that there is more than one such channel assignment, it choses one as follows. Let $S$ be the set of all channel assignments that traverse the smallest number of links. Let $H$ denote the number of such links. Now let $C_1$ be the set of all first channels of the channel assignments in $S$. Let $c_1$ be a channel in $C_1$ with wavelength that has the highest possible preference. If there is more than one then pick one arbitrarily. Let $S_1$ be the subset of channel assignments of $S$ that have $c_1$ as their first channel. Now for $k = 2, 3, ..., H$, let $C_k$ be the set of all possible $k^{th}$ channels of channel assignments in $S_{k-1}$. Let $c_k$ be a channel in $C_k$ with wavelength that has the highest possible priority. If there is more than one then pick one arbitrarily. Let $S_k$ be the subset of channel assignments of $S_{k-1}$ that have $c_k$ as their $k^{th}$ channel. The channel assignment returned by the Brute Force Search is one from $S_H$.

Here is an example of how Brute Force Search would select a channel assignment. Let $\{t_1, t_2, ..., t_7\}$ be the collection of shortest hop channel assignments, and let the wavelength preferences for the algorithm be Fixed. Let $\{t_1, t_3, t_7\}$ have their first channels at wavelength $w_0$, let $\{t_2, t_6\}$ have their first channels at wavelength $w_1$, and let $\{t_4, t_5\}$ have their first channels at wavelength $w_2$. Since $w_0$ is preferred, $S_1 = \{t_1, t_3, t_7\}$. Now suppose $\{t_3, t_7\}$ have their second channels at wavelength $w_1$, and $\{t_1\}$ has its second channel at wavelength $w_2$. Since $w_1$ is preferred, $S_2 = \{t_3, t_7\}$. Finally, suppose $t_3$ has its third channel at wavelength $w_2$, and $t_7$ has its third channel at wavelength $w_0$. Since $w_0$ is preferred, $t_7$ is the channel assignment returned by the Search.

The Brute Force Search, finds a channel assignment that uses the minimum number of channels. In addition, it tries to optimize the channel selection by choosing according to wavelength preferences.

The algorithm can be implemented as follows, which is similar to how *breadth first search* may be implemented [32]. The algorithm maintains a *queue*. Recall that a queue

is a data structure that has a *tail* and a *head*, data elements enter the tail and can be pulled from the head. The elements leave the head in the order they are put in at the tail. To implement a search of the channels, initially the queue is empty. Then all channels incident to the source node are pushed into the queue in order to the wavelength preference. While the destination has not been reached, a channel $c$ is pulled out of the queue. All channels $c'$ that are w-attached to $c$ that have not been searched yet, are pushed into the queue in order of their wavelength preference. In addition, it is recorded that $c'$ was found via $c$. We refer to $c$ as the *parent of* $c'$. The algorithm stops when it finds a channel $c$ incident to the destination node. The channel assignment can be found from $c$ by backtracking along the parents of channels.

The time complexity of Brute Force Search is the time complexity of classical breadth first search on the channel-graph. The number of vertices in the channel-graph is $O(WE)$, where $E$ is the number of fiber links and $W$ is the number of wavelengths. Thus, the time complexity of Brute force Search is $O(W^2E^2)$.

## 2.2 Networks With Minimal ADM/DCSs

We studied how to design networks at the tributary traffic layer that are cost efficient. We first studied ring networks. Figure 2.17 shows a node. It has *add-drop multiplexers* (ADMs) that multiplex and demultiplex tributary traffic onto lightpaths. An ADM is equivalent to two back-to-back LTEs. The node has a single DCS that cross-connects tributary traffic between ADMs and local ports. The node does not have any wavelength conversion. However, notice that the ADMs and DCS implement the wavelength conversion function since they switch traffic between wavelengths. The costs we considered are the ADMs and DCS. There are other costs such as the optical add-drop multiplexers, which multiplex and demultiplex wavelengths onto fibers. However, we assume that these costs are "absorbed" into the costs of the ADMs and DCSs.

Notice that cost savings occur whenever optical signals are allowed to *pass-through*, avoiding an ADM and its cost. This cost savings can be considerable in dense WDM networks with 80 or more wavelengths. However, optical pass-through means that the node has less switching capability. Tributary traffic that is on a pass-through wavelength cannot be switched to the node, nor is it able to change wavelengths at the node.

Our goal was to find networks that have optical pass-through but which have the

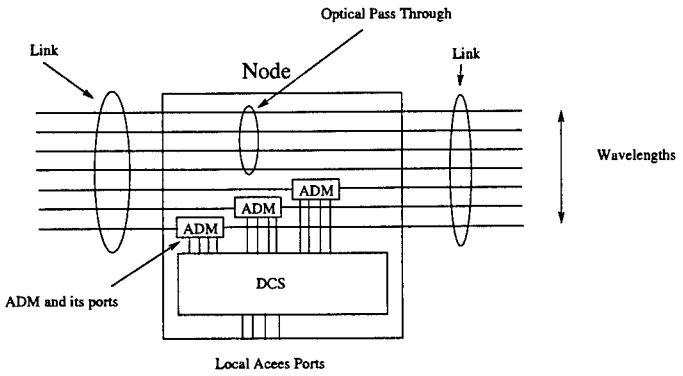



Figure 2.17: A node in a ring network.

same switching capability as a point-to-point network, i.e., a network with no optical pass-through. To explain what we mean by this we will describe what a point-to-point network can do. Consider a point-to-point ring network with $W$ wavelengths per link. All tributary traffic are *constant bit rate* (CBR) connections at fixed rates, and the number of them that can be multiplexed into a wavelength is denoted by $g$ (for granularity). For example, the optical line rate may be OC-48 (= 2.5 Gbps) while the tributary traffic rate is OC-3 (= 155 Mbps). Then $g = 16$ since OC-48 = 16× OC-3.

Note that each link may support $W \cdot g$ tributary traffic. To set up a traffic along a path in the ring, a necessary and sufficient condition is that there is free bandwidth on every link along the path. Note that the free bandwidth could be on different wavelengths because the traffic may be switched at intermediate nodes. In addition, the traffic may be set up without disturbing (rearranging) existing traffic.

Thus, as long as the tributary traffic load on every link is at most $W \cdot g$ no traffic will be blocked. Note that there are no statistical assumptions on the traffic. Traffic may arrive and depart at any time. Traffic may be routed in arbitrary ways. The only constraint is that the link loads are at most $W \cdot g$.

We would like to find a WDM network that has optical pass-through and has switching properties of a point-to-point network. However, this is impossible if the only constraint is on link loads. For example, consider the scenario where each link has $W \cdot g$ traffic, and each traffic connection traverses only one link. In other words, the links are completely filled with traffic, and each node must terminate all traffic from its incident fibers. Then, to insure no blocking, we must have an ADM at every wavelength to terminate the traffic. Thus, we need an additional constraint on traffic to guarantee no blocking and

have optical pass-through. The additional constraint we impose is to limit the amount of traffic that can terminate at any node. For example, each node $i$ could have a parameter $t(i)$, which is proportional to the amount of traffic that will terminate at the node, i.e., it is the maximum amount of bandwidth required by the node to access the network. In particular, the amount of traffic that may terminate at the node is at most $g \cdot t(i)$.

With such a constraint, if links are fully loaded with traffic then there will be a minimum amount of transit traffic through a node. This traffic, if *groomed properly*, can hopefully be transported on optical pass-through.

For this project, we defined a ring network architecture that is nonblocking for traffic that have link load constraints and node access constraints. We actually considered two traffic types which lead to different nonblocking properties for the network.

We developed a design algorithm for this network architecture. The design algorithm determines where to put ADMs (and optical pass-through) to minimize the cost and still keep the network nonblocking. We do not know if the solutions found by the algorithm are optimal but they do have much lower cost than the point-to-point network.

We also extended the ideas to general topology networks. We have a network architecture for general topology networks that requires some additional wavelengths over a point-to-point network. However, the number of additional wavelengths is small. We also provide a design algorithm.

## 2.3 Organization

The report is organized as follows. In Chapter 3, we discuss the models, assumptions, and performance measures we used. The chapter is divided into two parts. The first half is focused on limited wavelength conversion networks (i.e., the optical layer), and the second half is focused on the tributary traffic layer.

In Chapter 4, we present our results. Again, the chapter is divided into two parts. The first half is focused on limited wavelength conversion networks, and the second half is focused on the tributary traffic layer. We summarize our results and draw conclusions in Chapter 5.

We also provide in appendices, copies of three papers that were the outcome of this project.

**Appendix C:** O. Gerstel, R. Ramaswami, and G. Sasaki, "Cost effective traffic grooming in rings," *Proc. IEEE Infocom 98*, April 1998.

**Appendix D:** G. Sasaki, O. Gerstel, and R. Ramaswami, "A WDM ring network for Incremental traffic," *Proc.* $36^{th}$ *Annual Allerton Confernce on Communication, Control, and Computing*, Monticello, IL, Sept. 1998.

**Appendix E:** G. Sasaki and T. Lin, "A minimal cost WDM network for incremental traffic," *Proc. SPIE Conf. All-Optical Networking 1999: Architecture, Control, and Management Issues*, Boston, MA, Sept. 1999. It was a co-winner of the BEST paper award.

# Chapter 3

# Methods, Assumptions, and Procedures

In this chapter we will explain the type of methods, assumptions, and procedures that we used to evaluate our networks. We studied two switching technologies: limited wavelength conversion and minimal electronic ADMs/DCSs.

In the case of limited wavelength conversion the type of networks considered were at the optical layer. The traffic is lightpaths. We compared the performances of limited conversion with full conversion. Our method of performance evaluation was by simulation. In Section 3.1, we discuss our simulations including the traffic models, the kinds of statistics collected, and the sets of experiments that were conducted.

In the case of networks with minimal electronic ADMs/DCSs, the type of networks considered are at the tributary traffic layer. The type of traffic is tributary traffic that would be multiplexed into lightpaths. Thus, the traffic model here is finer grained than the lightpath traffic for limited wavelength conversion networks.

Our primary method to evaluate these networks was by mathematical analysis. We were able to show that the networks could guarantee no blocking of traffic under certain assmptions. These results are guarantees even for the worst case traffic scenario. As a result, the provisioning of network resources are conservative. Thus, we also ran simulations of random traffic to see how the networks perform for "typical" traffic. In addition, we also ran simulations to evaluate the efficiency of our network design algorithms. These traffic models and experiments are discussed in Section 3.2.

## 3.1 Limited Wavelength Conversion Networks

Our study of limited wavelength conversion focused on two aspects:

- Do the CAPs make a difference in performance? If they do then what CAPs perform best?
- What is the best way to manage the lightpaths?

In order to answer these questions, we ran computer simulations to evaluate the performance of the CAPs and channel assignment/management algorithms. Our investigation was organized as follows.

**Task A.1.** We studied the simple assignment algorithms described in Subsection 2.1.2. They are the Depth First Search (DFS) and Shortest Path (SP) algorithms. They are simple because they assume that lightpath requests come with their own pre-computed routes. Therefore, they only compute the wavelengths along the routes. In addition, they do not disturb existing lightpaths.

**Task A.2.** We determined which of the five types of CAPs discussed in Subsection 2.1.1 lead to the best performance. These are the three local CAPs (PARTITION, SHIFTED, and DISTRIBUTE) and the two expanding ones (RANDOM and SHUFFLE). To evaluate these CAPs, the better performing algorithms from Task A.1 were used.

**Task A.3.** We evaluated the other more complicated channel assignment/management algorithms. These include the ones that rearrange existing lightpaths in Subsection 2.1.3, and the ones for lightpaths without pre-computed routes in Subsection 2.1.4.

**Task A.4.** A design algorithm for determining good CAPs was developed and evaluated.

These tasks are explained in more detail in Subsection 3.1.3. In Subsections 3.1.1 and 3.1.2 we explain our network and traffic models, respectively.

### 3.1.1 Network Model

The network has a topology with the number of of nodes denoted by $N$, and the nodes numbered $1, 2, ..., N$. The primary topologies considered were the 16 node unidirectional ring and NSFNET as shown in Figures 3.1 and 3.2, respectively. The ring is an important topology because optical networks today are typically supported by the SONET/SDH

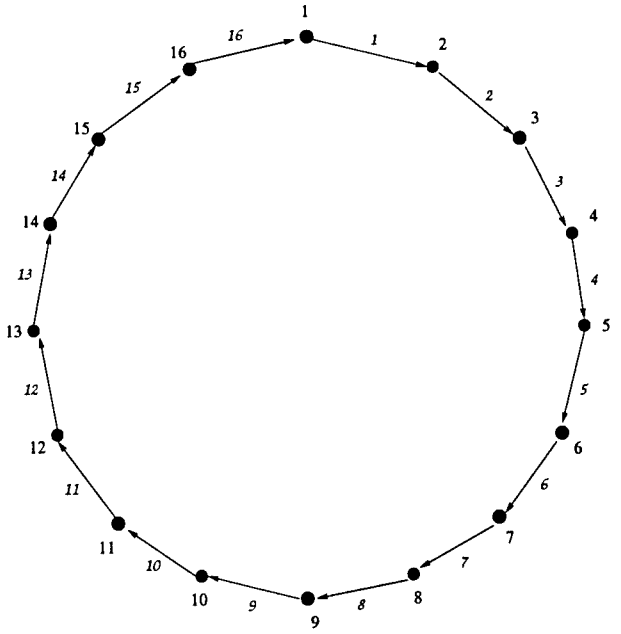



Figure 3.1: Ring with 16 nodes.

architecture, which relies on a ring topology to facilitate recovery from node and link faults. The NSFNET topology represents a mesh (and arbitrarily) connected topology. Note that for NSFNET in Figure 3.2, the numbers on the links are the "distances" of the links. For example, the distance could be the actual length or propagation delay.

We also considered other mesh topologies. REALNET, shown in Figure 3.3, is another example of a topology that is supposed to be "realistic" backbone network topology. The WORST-CASE topology, shown in Figure 3.4, is a 32 node topology where only the leftmost five and the rightmost five nodes source or sink traffic. This was done to increase the average path length of a lightpath. The WORST-CASE topology tends to put no wavelength conversion at a disadvantage.

A randomly generated topology RANDOM, shown in Figure 3.5 was also considered. The topology was constructed as follows. The number of nodes $N$ and number of edges $E$ were selected. As can be seen from the figure, $N = 14$ and $E = 19$. We randomly picked two distinct nodes, each pair being equally likely. If they had no edge then we added one. We kept doing this until $E$ edges were in the topology. Then we checked if the topology is connected. If not then we kept generating random topologies until we found a connected one.

With the exception of the 16 node unidirectional ring, all networks were assumed

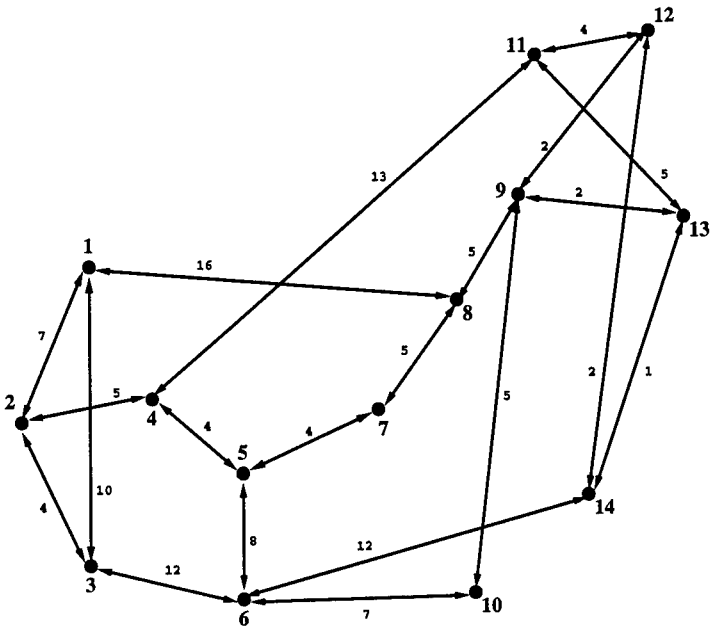



Figure 3.2: NSFNET.

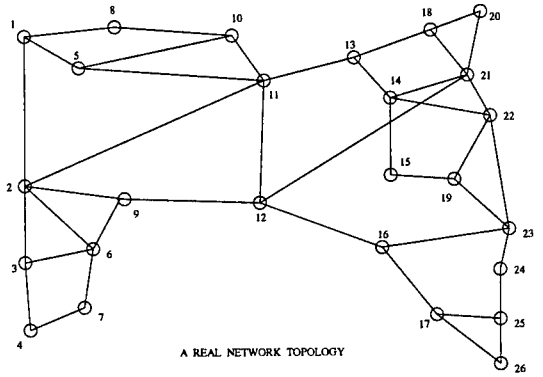
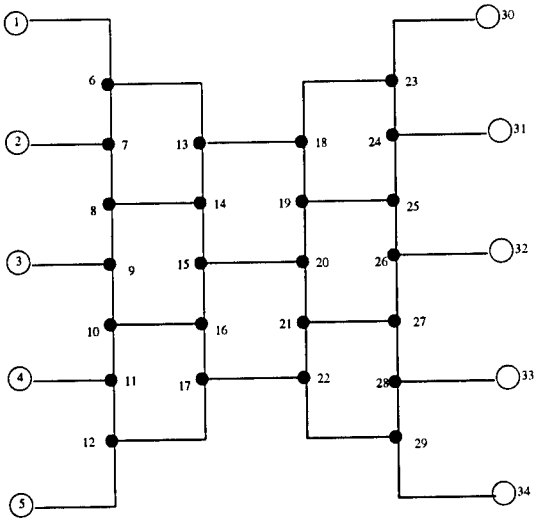
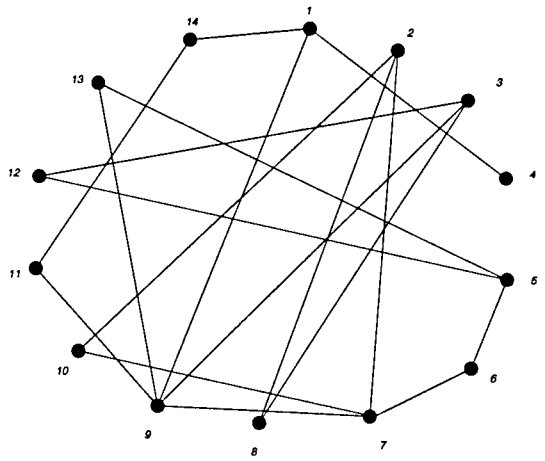



Figure 3.3: Another Real Network Topology (REALNET)




WORST CASE TOPOLOGY

Figure 3.4: The WORST-CASE topology




RANDOM TOPOLOGY

Figure 3.5: A Randomly Generated Topology (RANDOM)

to have *bidirectional* links. These bidirectional links are two unidirectional links going in opposite directions, where each unidirectional link corresponds to a fiber. The links employ WDM, where the number of wavelengths is denoted by $W$, and the wavelengths are $w_0, w_1, ..., w_{W-1}$. The wavelengths are ordered so that $w_0 < w_1 < ... < w_{W-1}$.

Note that the *unidirectional ring* can also model a bidirectional one. It models half a bidirectional ring, e.g., the half going clockwise (or counter-clockwise).

### 3.1.2 Traffic Model

We consider random *dynamic* and *semi-dynamic* (or *incremental*) lightpath traffic.

**Random Dynamic** For this type of traffic, lightpath requests arrive and terminate at random times. In particular, lightpath requests arrive as a Poisson arrival process. The requests do not have pre-computed routes. The source and destination nodes of the lightpath requests are determined randomly according to a traffic matrix $\{t_{i,j}\}$, where $t_{i,j}$ specifies the arrival rate of lightpath requests with source node $i$ and destination node $j$. Thus, the lightpath requests between a pair of nodes $i$ and $j$ arrive as a Poisson arrival process with rate $t_{i,j}$. The holding times of all lightpaths are independent and exponentially distributed with some fixed mean. Naturally, the termination time of a lightpath is the sum of its arrival and holding times.

The traffic model is a classical one that has been used to evaluate circuit switched networks, e.g., telephone networks. It has also been used to evaluate WDM networks (e.g., [22]). In most of the other studies, the performance of a network was measured by the blocking probabilities of lightpath requests. Moreover, it was assumed that the operating point of the networks was low enough to keep the blocking probabilities small. Low blocking probabilities are desirable because a network should be designed so that it is improbable that a new lightpath will be rejected. However, the low operating point means that the links are under-utilized. This is unrealistic because WDM equipment is expensive, so a network will be provisioned to make use of most of its bandwidth.

We use another performance metric, the *time to first block*. For this metric, we assume the network is empty at the initial time 0. The *time to first block* is just the first time that an arriving lightpath request is blocked by the network. The time to first block is a performance metric that would be of interest to telecommunication network companies

because it indicates when the company should provision more resources. In addition, with this metric, we do not have to be careful about choosing arrival rates and mean holding times to keep the utilization low. For example, we could have relatively long holding times, which would be appropriate for very high bandwidth connections, such as lightpaths, that carry many tributary traffic and are unlikely to be terminated soon.

**Random Semi-Dynamic (or Incremental) Traffic** For random semi-dynamic traffic, lightpath requests arrive at random times, but their holding times are infinite. This models high bandwidth lightpaths (say at OC-192 rates, 10 Gbps) that carry many tributary traffic streams (say at OC-3, 155Mbps, or lower). Thus, they are unlikely to be terminated in the near future.

This traffic model assumes that a finite set of lightpath requests arrive in network, where each request comes with its own pre-computed route, and the total number of lightpath requests over any link is exactly $W$. Note that this traffic model sends a high lightpath load to every link. However, the lightpath load does not exceed the capacity of any link. In fact, if the network had full wavelength conversion at every node, the lightpath request set could be completely accommodated. In other words, there would be no blocking of the lightpath requests. Thus, if the network had limited wavelength conversion, the number of lightpath requests that are blocked is a measure of the degradation of performance of limited conversion compared to full conversion.

The traffic model considers lightpath requests in sets (or *batches*), where each set has exactly $W$ requests over each link. The performance metric is the *blocking probability*, i.e., fraction of blocked requests. If the blocking is zero then the network performes as well as a network with full conversion. In order to measure the performance metric, we generate multiple batches of lightpath requests. Each batch corresponds to a simulation run, starting from an initially empty network. During a simulation run, the following is done.

1. Generate a random batch of routes for lightpaths such that each link has exactly $W$ routes going in either direction.

2. Randomly order the routes.

3. Have the routes arrive into the network as lightpath requests and as a Poisson

arrival process according to their order. Set up the lightpaths and if the set-up fails, consider the lightpath *blocked.*

In the simulations, the arrival rate is equal to one. However, note that the exact value of the arrival rate is unimportant because the performance metric just depends on the number of blocked lightpaths and lightpaths are never terminated

In order to determine a blocking probability for the network, multiple simulation runs (each corresponding to a different batch) were executed. The blocking probability is equal to the total number of blocked lightpaths divided by the total number of lightpath arrivals.

We now describe how a batch of lightpath requests are computed. There are two methods.

- *Unidirectional ring.* This method of computing a batch of lightpath requests is for the unidirectional ring only. The routes of the lightpath requests are computed as follows. We will assume the ring goes in the clockwise direction. A node is randomly chosen as the *initial node*, which we denote by $x_0$. The first route starts at $x_0$, goes clockwise, and has length that is random and uniformly distributed on the interval $\{1, ..., \lfloor \frac{N}{2} \rfloor\}$. (Note that the choice of the route's length models shortest hop routing on the bidirectional ring.) Let the terminating node be denoted by $x_1$. The second route starts at $x_1$, goes clockwise, and has length that is random and uniformly distributed on the interval $\{1, ..., \lfloor \frac{N}{2} \rfloor\}$. Let the terminating node be denoted by $x_2$. The routes are continually generated this way, until a route is generated that crosses node $x_0$ for the $W^{th}$ time. Then that route is truncated at $x_0$, and the batch of routes is complete. Note that the batch traverses each link exactly $W$ times.

- *General topology.* This method of computing a batch of lightpath requests can be applied to any topology including the unidirectional ring. The method finds a set of routes for the lightpath requests one at a time until each link has exactly $W$ lightpath requests. The computation of a lightpath request is as follows. For each node $s$, let $T_s$ be the set of nodes, other than $s$, that can be reached by $s$ by following links, that have less than $W$ lightpath requests.

To create a lightpath request, randomly choose a node $s$ such that $T_s$ is not empty, each possible node being equally likely. Node $s$ is the source of the lightpath request. Randomly choose a node $t$ from $T_s$, each being equally likely. This is the destination of the lightpath request. The route of the request is just a shortest distance route from $s$ to $t$ along links with less than $W$ lightpath requests. With the exception of NSFNET (see Figure 3.2), all topologies have the distances of their links equal to one. For NSFNET, the link distances are given in Figure 3.2.

### 3.1.3 Simulation Details

We will list the details about simulations for the four tasks.

- **Task A.1.** We evaluated the DFS and SP channel assignment algorithms. All variations of DFS and SP were considered. In particulared, for DFS, the wavelength preferences of Fixed, Random, Pack, and Spread were evaluated. For SP, different channel cost functions were considered.

  The simulations of the algorithms were for different numbers of wavelengths, five types of CAPs (DISTRIBUTE, PARTITION, SHIFTED, SHUFFLE, and RANDOM), different w-degrees (2 and 4), and different topologies, though primarily the 16 node unidirectional ring and NSFNET. For the ring topology, the random semi-dynamic traffic model for the unidirectional ring was used. For the other topologies, the random semi-dynamic for general topologies was used.

  The performance metric was the probability of blocking. The probabilities were computed after 1000 (or more) batches of lightpath requests.

- **Task A.2.** For this task, we evaluated five types of CAPs, which were the three local ones (DISTRIBUTE, PARTITION, and SHIFTED) and the two expanding ones (RANDOM, SHUFFLE). We used the semi-dynamic traffic models as in Task A.1. The topologies were the ring and NSFNET. The performance metric was the blocking probability, and it was computed after 1000 (or more) batches of lightpath requests. The channel assignment algorithm used was DFS with wavelength preferences of Fixed and Pack since it was shown in Task 1 that it worked well.

- **Task A.3.** For this task, we evaluate two types of channel assignment algorithms. The first type were the ones that can rearrange exising lightpaths. These are the

RAS, Single-MTV, and Multiple-MTV algorithms. They assume that lightpath requests come with their own pre-computed routes. The second type of channel assignment algorithms are ones that do not assume that lightpath requests come with pre-computed routes. These algorithms are Widest Shortest, Shortest Distance, and Brute Force Search.

In the case of RAS, Single-MTV, and Multiple-MTV, the traffic model was semi-dynamic for general topologies. This model was used even with the ring topology. The performance metric was the blocking probabilities after 1000 batches of lightpath requests. For these channel assignment algorithms, the DFS with Fixed wavelength preferences was used as a subroutine to set up lightpaths along routes. In all experiments, local CAPs were used since it was shown in the experiments of Task 2 that they performed better than expanding CAPS.

The second type of channel assignment algorithms are the ones that assume no pre-computed lightpath routes. They are Widest Shortest, Shortest Distance, and Brute Force Search. The traffic model used was random dynamic traffic. The traffic matrices were generated randomly. Each pair of source and destination nodes $(i, j)$, was assigned a random integer $t_{ij}$, independently of the others, which was uniformly distributed from 0 to 14. (The number 14 is an arbitrary value.) For all topologies, except the WORST-CASE (Figure 3.4), each possible (ordered) pair of nodes is a source-destination pair and assigned such a random rate. For the WORST-CASE topology, as shown in Figure 3.4, the only pairs of nodes that are source-destination pairs are those that have a source on the left side and the destination on the right side and vice versa. Thus, all lightpaths cross the topology from the left side to the right or vice versa.

The mean holding time of a lightpath was 0.35. The performance metric was the average time to first block. The averages were taken over 1000 simulation runs.

- **Task A.4.** To implement DesignCAP, we need a *cost function* $f(c)$ that is a function of CAPs $c$. This cost function should reflect the performance of the CAP $c$. The way we implement this is to run simulations to estimate this performance. The simulation is on a small 8 node unidirectional ring network. The network is small enough so that the simulations do not take too much time, but the lightpaths

are long enough so that wavelength conversion is important for efficient bandwidth utilization.

The simulations used a modification of the semi-dynamic traffic for unidirectional rings, described in Subsection 3.1.2. The modification creates longer lightpaths as follows. In the original semi-dynamic traffic for unidirectional rings, a lightpath route is created by starting at some source node and then picking a route that has length that is random and uniformly distributed between 1 and $\lfloor N/2 \rfloor$. This has been modified for DesignCap so that the length is uniformly distributed between 1 and $N-1$.

To get a value for $f(c)$, 10,000 batches semi-dynamic lightpath requests were simulated on an 8 node unidirectional ring, where each node has CAP $c$. The channel assignment algorithm was DFS with Fixed wavelength preferences. The value of $f(c)$ is the average number of lightpath requests that were blocked in a batch. Note that the value of $f(c)$ is random since it is an outcome of the simulations. Simulating more batches would make $f(c)$ less variable, but at the expense of longer computing times.

### 3.1.4 Simulation Implementation

The simulators used for the experiments were all written in C and compiled on a gcc GNU project C compiler. The simulator is designed to run on a Sun UltraSPARC-1 workstation using the SunOS 5.6 operating system.

## 3.2 Networks with Minimal ADM/DCSs

Our study of networks at the tributary traffic layer was focused on cost efficient designs. The network designs must support tributary traffic streams so that the network is bandwidth efficient. At the same time, the network must allow optical pass through as much as possible to avoid the cost of ADMs, LTEs, and DCSs. Our study was organized into the following tasks.

- **Task B.1.** We investigated ring network architectures. Ring topologies are important because of the SONET/SDH architecture. Our task was to determine network

architectures that insure that there is no blocking of tributary traffic. The traffic models we used impose constraints on the amount of traffic that may terminate at a node. This is an access-bandwidth constraint on the node to the network. Some of the models also have a traffic load constraint on links. We were successful in finding a set of network architectures. One of them, called the *Incremental* ring, is as bandwidth efficient as a point-to-point network.

For the Incremental ring, we developed a design algorithm that places ADMs and DCSs to minimize cost. We tested the design algorithm by running it on randomly generated instances.

- **Task B.2.** We investigated general topology networks. The investigation was similar to our study for rings. Our task was to find a network architecture that insures that there is no blocking of tributary traffic. The traffic model we used is similar to the one used in Task B.1 for rings. We did find a network architecture that is almost as bandwidth efficient as a point-to-point network. We developed a design algorithm for this architecture. The algorithm determines where the LTEs, DCSs, and OXCs should be placed. We tested the design algorithm by running it on randomly generated instances.

In Subsection 3.2.1 we describe the network models for the rings and general topologies. In Subsection 3.2.2 we describe our traffic models. There are two types of traffic models. The first model is a set of constraints that the traffic must conform to. These constraints lead to guarantees of no blocking. The second model is for random traffic to determine the performance of the networks for random "typical" traffic. For this model, statistical assumptions are given. The first model is also used to generate random instances to evaluate network design algorithms. In particular, the network design algorithms are dependent on parameters of the first traffic model. We describe how we randomly choose these parameters to evaluate our design algorithms.

### 3.2.1 Network Model

There are two types of topologies that we considered: ring and the general (or arbitrary) topology. $N$ denotes the number of nodes in the network, where the nodes are numbered $0, 1, ..., N-1$. (Note that the nodes are numbered differently than for limited wavelength

conversion networks that have nodes numbered $1, 2, ..., N$.) For the ring, the nodes are numbered in increasing order going clockwise. The number of wavelengths in the network is denoted by $W$.

The links in the network are bidirectional. They each have $W$ WDM bidirectional channels. Note that, at a node, the signals of a WDM channel are either terminated by an ADM, or are passed through to the WDM channel on the next link. Each WDM channel can carry $g$ tributary bidirectional traffic streams for some integer $g$. For example, the line rate for a WDM channel may be OC-48 (2.5 Gbps) and a tributary stream may be OC-3 (155 Mbps). Then $g = 16$ because $OC - 48 = 16 \times OC - 3$.

For the ring network, a node $i$ has a collection of ADMs and a single DCS as shown in Figure 2.17. The DCS cross-connects the traffic streams between the ADMs and local ports. The DCS is assumed to be *wide sense nonblocking*, which means that it can switch a traffic stream without disturbing existing streams. The node has an integer parameter $t(i)$ which indicates the maximum number of local ports at the node. This parameter is explained in the next subsection on traffic models.

The cost of the node is assumed to be composed of the cost of ADMs and the DCS. The cost of the ADMs is proportional to the number of them. We denote the number of ADMs at node $i$ by $A(i)$. The cost of the DCS is a function of the number of its ports. We assume that the number of its ports is proportional to $A(i) + t(i)$, which in turn is proportional to the number of ports from the ADMs and the local ports. For example, the cost of the DCS could be a quadratic function of $A(i) + t(i)$. In any case, we assume that the cost of the ADMs and the DCS at a node $i$ is some function $D(A(i), t(i))$, which is dependent on $A(i)$ and $t(i)$.

For a general topology network, there are two types of nodes: *opaque* and *optical*. An opaque node is shown in Figure 3.6. Each WDM channel is terminated by an LTE and there is a single DCS to cross-connect tributary traffic between the LTEs and local ports. An opaque node is the same type of node in a point-to-point network.

An optical node is shown in Figure 3.7. It has LTEs, a single DCS (to cross-connect all tributary traffic), and *optical cross-connects* (OXCs). The OXCs cross-connect pass-through optical signals. The OXCs correspond to distinct wavelengths, though not all wavelengths will have an OXC. At a node $i$, wavelengths are partitioned into two types: *transit* and *local*. Transit wavelengths allow optical pass-through. Each transit wave-

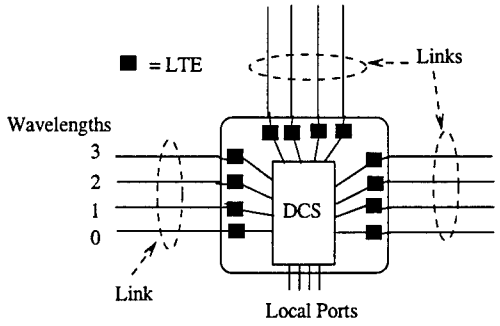



Figure 3.6: An opaque node.

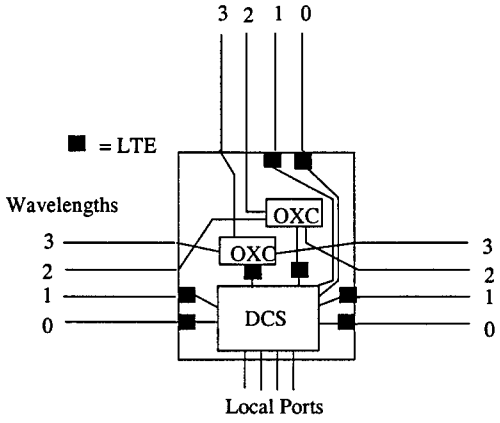



Figure 3.7: An optical node.

length has $d(i) - 2$ LTEs, where $d(i)$ is the number of incident bidirectional links at node, i.e., the node's *degree*. Each transit wavelength also has an OXC that will cross-connect the optical signals between all incident fiber links and the LTEs for the wavelength. Now at each local wavelength, there are LTEs that terminate all the WDM channels.

The cost of a node is comprised of the cost of the ADMs, DCS, and OXCs. Let $A(i)$ be the number of ADMs at a node $i$. Let $t(i)$ be a parameter that limits the amount of traffic at the node. (This will be explained in the next subsection on traffic models.) Then the cost of the ADMs at node $i$ is assumed to be $A(i)$, i.e., unit cost per ADM. The cost of an OXC is proportional to the square of the number of its ports. Note that it has a port per incident link and a port for each of the $d(i) - 2$ LTEs for the corresponding transit wavelength. Thus, the cost is $b \cdot (2d(i) - 2)^2$, for some constant $b$. The DCS cost is a function of the number of its ports. We will assume that the number of its ports is proportional to the sum of $A(i)$ and $d(i) \cdot t(i)$, where the the second term is a bound on the maximum number of tributary traffic that may terminate at node $i$. If the DCS cost

is proportional to the square of its ports then it is $a \cdot (A(i) + d(i) \cdot t(i))^2$, where $a$ is some constant.

For both ring and general topologies, the networks we discovered do not block traffic, under certain traffic models. In other words, the networks were *nonblocking*. There are two types of nonblocking networks: *rearrangeably nonblocking* and *wide-sense nonblocking*. Rearrangeably nonblocking means that existing traffic streams may be rearranged to make way for a new traffic. Wide-sense nonblocking means that existing streams are not disturbed.

### 3.2.2 Traffic Models

We used two types of traffic models.

- Non-statistical dynamic traffic for nonblocking results.
- Random dynamic traffic to evaluate the networks under "typical" traffic.

We will first describe these traffic types for rings, and then for general topologies.

**Rings: Non-Statistical Dynamic Traffic for Nonblocking Results** The traffic is *dynamic*, which means that tributary traffic streams may arrive and depart at arbitrary times. We will also consider *semi-dynamic* (or *incremental*) traffic, which is dynamic traffic, but where traffic never terminates.

The traffic distribution will be represented by a matrix $T = \{T(i,j)\}$, where $g \cdot T(i,j)$ equals the number of traffic streams beween nodes $i$ and $j$. Thus, $T(i,j)$ is the number of "lightpaths of traffic" between nodes $i$ and $j$. Note that $T(i,j)$ can be fractional. For example, if 24 OC-3 connections are to be supported between $i$ and $j$ and $g = 16$ then $T(i,j) = 1.5$.

The routing of traffic affects the traffic loads on links, which in turn affect bandwidth requirements. We consider traffic that either requires routing or are *pre-routed*, i.e., they come with their own pre-computed routes. In addition, pre-routed traffic are assumed to have *simple* routes, which means that they visit a node at most once. In this sense, they are routed efficiently in the network. Note that the pre-routed traffic model holds for many practical scenarios, such as when traffic is routed according to shortest paths or

traffic-loads. It allows us to define a maximum "traffic load" over links, which is a lower bound on the number of wavelengths to accommodate the traffic to insure no blocking.

We consider two different traffic assumptions (i.e., scenarios) that are given below. The first assumes dynamic traffic that only has retrictions on the amount of traffic that terminates at the nodes. The next assumption has pre-routed traffic and a maximum traffic link load parameter. The parameter is a measure of the required bandwidth (wavelengths) on the links. This model may be more appropriate when wavelengths are limited because then the load parameter value can be chosen appropriately.

- **Traffic Assumption B.1.** Traffic is dynamic, i.e., $T$ is time-varying. The traffic has integer parameters $(t(i) : i = 0, 1, ..., N-1)$. At any time, each node $i$ can source/sink at most $g \cdot t(i)$ traffic streams. Thus, at any time, for each node $i$, $t(i) \geq \sum_{j=0}^{N-1} T(i,j)$ and $t(i) \geq \sum_{j=0}^{N-1} T(j,i)$. □

- **Traffic Assumption B.2.** The traffic is dynamic, with traffic streams being pre-routed and having simple routes. The traffic has integer parameters $L$ and $(t(i) : i = 0, 1, ..., N-1)$. At any time, the number of traffic streams over any link is at most $g \cdot L$, assuming no blocking. In addition, each node $i$ may source/sink at most $g \cdot t(i)$ traffic streams from the clockwise or counter-clockwise direction along the ring. Thus, node $i$ can terminate up to $2g \cdot t(i)$ traffic streams, but then half must come from the clockwise direction and the other half must come from the counter-clockwise direction. (Note that $t(i)$ is a lower bound on the number of ADMs at node $i$ to insure no blocking.)

- **Traffic Assumption B.3.** It is Assumption B.2 and the following additional assumption. For each node $i$, the amount of transit traffic is at most $g \cdot (L - t(i))$.

We have network architectures architectures that guarantee no traffic is blocked under the traffic assumptions. These are described in Subsection 4.2.1. The dimensions and cost of the network will be dependent on the parameter values of the traffic model, and in particular $L$ and $\{t(i)\}$. For one of the ring architectures, called the *Incremental* ring , we developed a design algorithm that minimizes its cost. To evaluate this design algorithm, we use a method to find "typical" random values for $\{t(i)\}$. We could then measure an average cost of a network by running the design algorithm of multiple sets on randomly generated $\{t(i)\}$.

We now describe the procedure to generate a random collection of $\{t(i)\}$. First, the procedure depends on a parameter $F$. This parameter controls the variation of the values of $\{t(i)\}$. The values of $\{t(i)\}$ have more variation if $F$ is larger. Given $F$, the procedure randomly computes a batch of traffic. From this batch, the $\{t(i)\}$ values are determined.

We now describe how a batch of traffic and then $\{t(i)\}$ are computed. First, random and statistically independent numbers are computed for each node $i$, where $f_i$ denotes the number. The $f_i$ are uniformly distributed on the interval $[1, F]$. $f_i$ indicates the "frequency" that node $i$ sources/sinks a traffic. For example, if $f_i/f_j = 2$ then we would expect $i$ to source/sink twice as much traffic as node $j$.

Given the $\{f_i\}$, we generate a random batch of traffic similar to random semi-dynamic traffic for limited wavelength conversion, unidirectional ring networks. Each link will have exactly $g \cdot L$ traffic streams. We start from some arbitrary node, say node 0. Then traffic streams are generated going in the clockwise direction around the ring. Each stream (except possibly the last) is generated follows. It starts where the last stream ended. It is routed clockwise and its end node will be at most $N/2$ hops away (to model shortest path routing). Its end node is chosen randomly according to $\{f_i\}$. Thus, node $i$ will be $f_i/f_j$ more likely to be chosen than node $j$. However, there is an exception for the node that is exactly $N/2$ hops away. Its frequency is decreased by half, to take into account that a shortest hop path between two nodes at exact opposite ends of the ring could be on either side of the ring. The last stream occurs when each link has at least $g \cdot L$ traffic streams. The last stream is truncated so that each link has exactly $g \cdot L$ traffic streams. Then for each node $i$, $t(i) = \lceil h(i)/g \rceil$, where $h(i)$ is the number of traffic streams sourced/sunk at node $i$ from either link.

**Rings: Dynamic Traffic To Evaluate The Networks Under "Typical" Traffic**
We evaluated the Incremental ring architecture with traffic that conformed to Assumption B.2, but is random. The traffic is a modified version of another traffic, which refer to as the *original Poisson traffic.* The modified traffic will be referred to as *conforming* to Assumption B.2.

The original Poisson traffic has traffic streams arriving as a Poisson process at rate $a \cdot g \cdot N \cdot W$, where $a$ is a parameter. An arriving stream has one terminating node that is randomly chosen, uniformly distributed on $[0, N-1]$. The route of this stream goes

clockwise from the node and has a random length that is nearly uniform on the interval $[1, N/2]$. In particular, if the probability of the length being $h$ is denoted by $p_h$, then $p_1 = p_2 = ... = p_{N/2-1} = 2p_{N/2}$, where $N$ is assumed to be even. For example, if $N = 8$ then $p_1 = p_2 = p_3 = 2/7$ and $p_4 = 1/7$. This models shortest hop routes, where routes have length at most $N/2$. Also, for the original Poisson traffic, the traffic streams have random holding times that are exponentially distributed with mean of one time unit.

The original Poisson traffic will not necessarily conform with Assumption B.2. Thus, the traffic will be modified by discarding arriving streams that will violate the assumption. In particular, if an arriving stream will cause a load on a link to exceed $g \cdot L$ then it is discarded. Also, if an arriving stream will cause one of its end nodes $i$ to have more than $g \cdot t(i)$ terminating traffic streams then it is discarded. The undiscarded traffic streams satisfies Assumption B.2 and is referred to as *conforming* to Assumption B.2. The conforming traffic was applied to our simulations.

**General Topology: Nonstatistical Dynamic Traffic For Nonblocking Results**
The assumptions are similar to the ones for ring networks.

- **Traffic Assumption B.4.** The traffic is dynamic with traffic streams being pre-routed and having simple routes. The traffic has integer parameters $L$ and $(t(i) : i = 0, 1, ..., N-1)$. At any time, the number of traffic streams over any link is at most $g \cdot L$, assuming no blocking. In addition, each node $i$ may source/sink at most $g \cdot t(i)$ traffic streams through any of its incident links. Thus, node $i$ can terminate up to $g \cdot t(i) \cdot d(i)$ traffic streams, where $d(i)$ is the degree of $i$. However, at most $g \cdot t(i)$ can be come through any link.

We have a network architecture that guarantees that no traffic is blocked under the traffic assumption which is described in Subsection 4.2.2. The dimensions and cost of the network will be dependent on the parameter values of the traffic model, and in particular $L$ and $\{t(i)\}$. We developed a design algorithm that minimizes the cost. To evaluate this design algorithm, we use a simple method to find "typical" random values for $\{t(i)\}$. In particular, to generate random $\{t(i)\}$ values, we randomly and independently generate them so that they were uniformly distributed over a given interval. We could then measure an average cost of a network by running the design algorithm on multiple sets of randomly generated $\{t(i)\}$.

**General Topology: Random Dynamic Traffic To Evaluate The Networks Under "Typical" Traffic** To evaluate the "typical" performance of the general topology network architecture, we simulated it with random traffic. The traffic is similar to the Poisson traffic that conforms to Assumption B.2. The traffic is modified from an *original Poisson traffic.* We refer to this modified traffic as *link load conforming.* The link load conforming traffic will have at most $g \cdot L$ traffic over every link, where $L$ is the same parameter as in Assumption B.4.

The original Poisson traffic has traffic arriving as Poisson streams at some aggregate rate. The traffic comes with their own routes, i.e., they are pre-routed. The holding times of the traffic are random and exponentially distributed with some mean.

The original Poisson traffic may have more than $g \cdot L$ traffic streams over a link. To get traffic that is link load conforming, an arriving stream that loads some link with more than $g \cdot L$ streams is discarded. Thus, the undiscarded traffic is *link load conforming.* This traffic was used in our simulations.

# Chapter 4

# Results and Discussion

We will present the results of this project. In Section 4.1, we have our results for limited wavelength conversion networks. These networks are in the optical layer supporting dynamic lightpaths. The results accomplish Tasks A.1-A.4. In Section 4.2, we have our results for networks with minimal ADMs (LTEs), DCSs, and OXCs. These networks are in the tributary traffic layer. The results accomplish Tasks B.1 and B.2.

## 4.1 Limited Wavelength Conversion Networks

We will present results of Tasks A.1-A.4 discussed in the previous chapter. Section 4.1.1 has our experimental results for Task A.1 which was to evaluate the simple channel assignment algorithms DFS and SP. The results for Task A.2 are in Section 4.1.2. Here we describe the performance of local and expanding CAPs. Sections 4.1.3 and 4.1.4 has the results for Task A.3. Section 4.1.3 evaluates channel assignment algorithms that allow lightpath rearrangement. Section 4.1.4 evaluates algorithms that do not assume lightpath requests come with pre-computed routes. The results of Task A.4 are presented in Section 4.1.5. This is the simulation results of DesignCAP.

### 4.1.1 Simple Channel Assignment

The performance of the DFS and SP channel assignment algorithms will be discussed. The simulation results are summarized below:

- The performance of DFS will be evaluated with the different wavelength preferences, Fixed, Random, Pack, and Spread. The simulations were done for the ring and NSFNET topologies. It is shown that Fixed and Pack perform best.

- The performance DFS and SP are compared next. For DFS, Pack and Fixed were used. For SP, different cost functions were used. It is shown that DFS and SP work about the same. DFS with Fixed leads to the best performance.

We will describe the simulation results for DFS first. There are four variations of this algorithm that depend on the wavelength preference used (i.e., Fixed, Random, Pack, or Spread). It turns out that Fixed and Pack perform better than Random and Spread for the 16 node ring network. Figures 4.1, 4.2, and 4.3 show the blocking probabilities for different numbers of wavelengths $W$ (ranging up to 32) for the local CAPs PARTITION, SHIFTED, and DISTRIBUTE, respectively, for w-degree 2. The curves corresponds to Fixed, Randm, Pack, and Spread. For the sake of comparison, a curve is given that corresponds to no conversion and using Pack.

For the case of SHIFTED, the even numbered nodes had the PARTITION CAP, while the odd numbered nodes had the S-PARTITION CAP. Also notice that the figures have the blocking probability for the case when the nodes have no conversion and Pack is the wavelength preference.

The figures show that Fixed and Pack performed better than Random and Spread, with Fixed being a little better than Pack. This implies that trying to pack lightpaths in the most heavily used wavelengths tends to free the other wavelengths for future lightpaths. Note that Random and Spread perform so poorly that they have higher blocking probabilities than a network with no conversion and Pack.

These results correlate with the conclusions given in [22], where it was shown that Fixed and Pack perform about the same, and are better than Spread and Random. Our results differ slightly because Fixed works a little better than Pack, while the opposite is true in [22]. We believe this is due to the differences in traffic model. The traffic model in [22] has finite holding times, and is therefore more dynamic in how traffic load is distributed among the wavelengths. Pack will perform better in this case because it is more robust to the traffic distribution.

Figures 4.4, 4.5, and 4.6 show blocking probability curves for the 16 node ring network, w-degree 2, and the expanding CAPs: RANDOM-SAME, RANDOM-RANDOM, and SHUFFLE, respectively.

For the expanding CAPs, both Fixed and Pack have blocking probabilities that are not much worse than Random and Spread, and sometimes much better. However, these

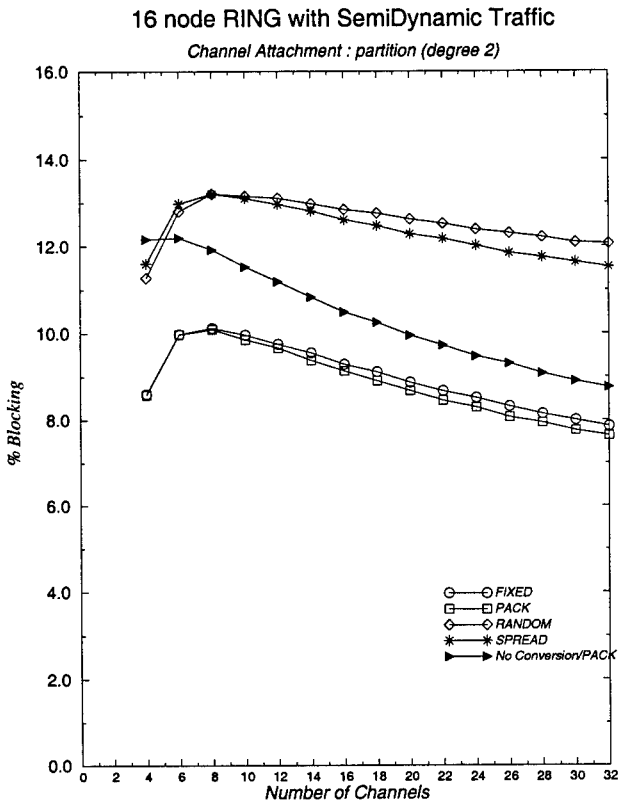



Figure 4.1: Blocking probabilities for the 16 node ring, w-degree 2, DFS, and PARTITION.

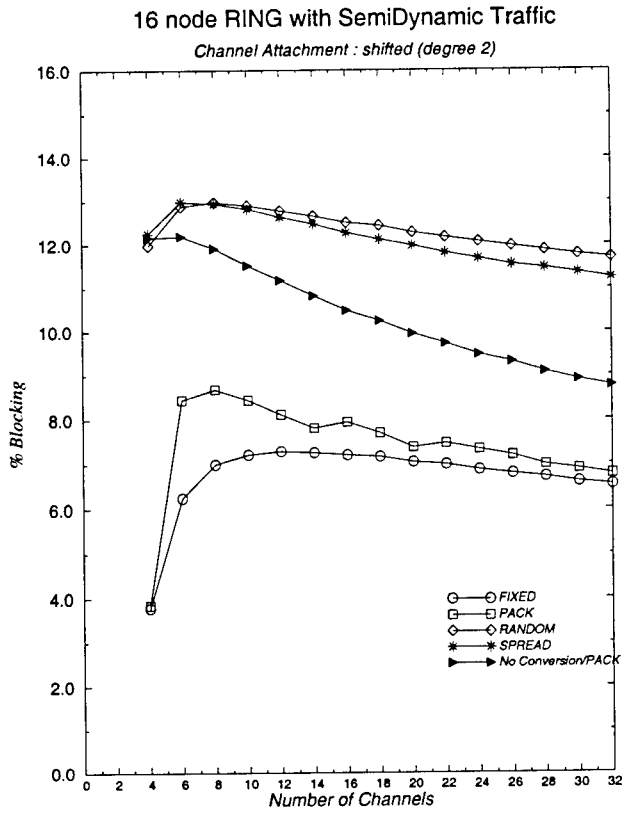



Figure 4.2: Blocking probabilities for the 16 node ring, w-degree 2, DFS, and SHIFTED.

CAPs have higher blocking probabilities than the local ones. To illustrate this, note that they lead to higher blocking probabilities than for a ring with no conversion (using Pack). Our explanation is that the expanding CAPs lead to lightpaths being distributed over all the wavelengths, breaking up trails of free channels. This makes it more difficult for future lightpaths to find a channel assignment.

We also conducted experiments for the NSFNET topology for w-degree 2. The plots of blocking probabilities for the different wavelength preferences and CAPs are left in Appendix A. The results are the same as for the ring. We can conclude that for the DFS channel assignment, Pack and Fixed work about the same and better than Random and Spread.

For the rest of this section, we will report on simulations for shortest path channel assignment and compare it with DFS. For both channel assignment strategies, the measured channel utilization values will be used. In the case of shortest path, the channel utilization $u_i$ for each wavelength $w_i$ is used. In particular, the *cost* of a channel with wavelength $w_i$ is $c_i = (1 - u_i/u_{max})^a$, where $u_{max} = \max_k u_k$ and $a$ is some parameter.

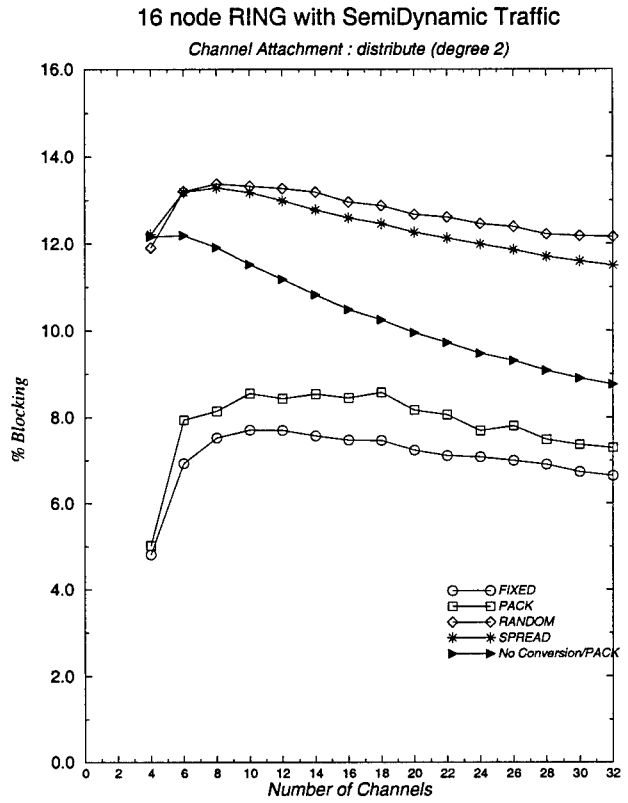



Figure 4.3: Blocking probabilities for the 16 node ring, w-degree 2, DFS, and DISTRIBUTE.

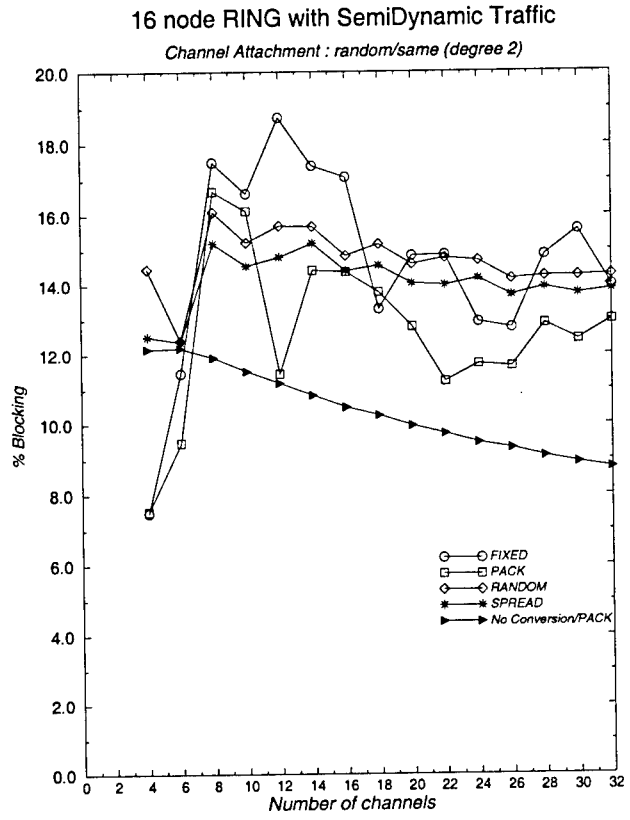



Figure 4.4: Blocking probabilities for the 16 node ring, w-degree 2, DFS, and RANDOM-SAME.

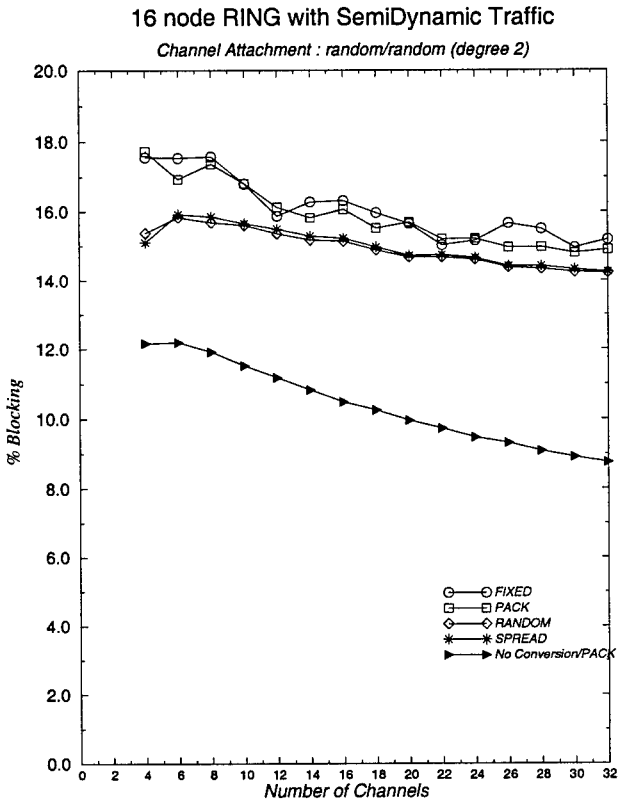



Figure 4.5: Blocking probabilities for the 16 node ring, w-degree 2, DFS, and RANDOM-RANDOM.

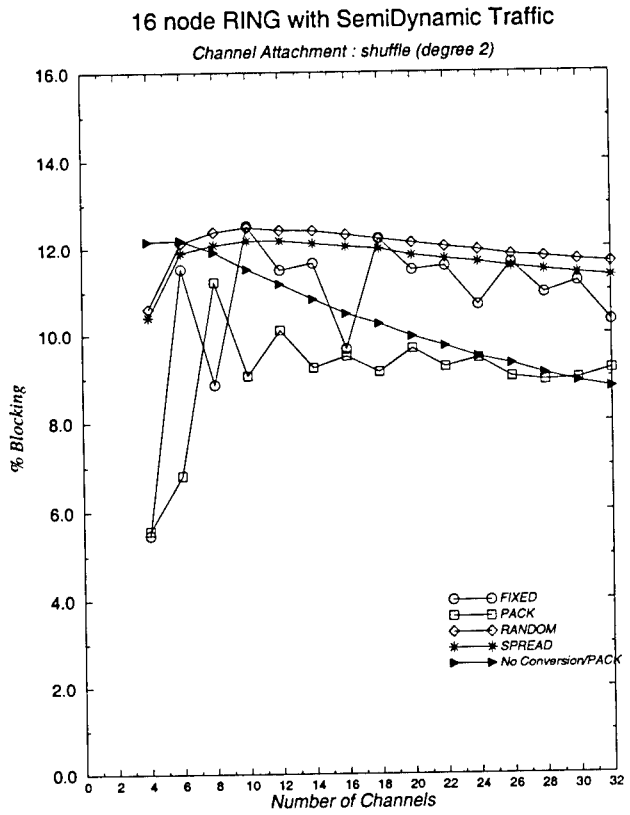



Figure 4.6: Blocking probabilities for the 16 node ring, w-degree 2, DFS, and SHUFFLE.

Thus, channels with smaller cost correspond to wavelengths with larger utilizations.

Figure 4.7 presents blocking curves for three versions of the shortest path channel assignment, corresponding to the three values of $a = 1, 50$, and 500. The network topology is the 16 node ring, $W = 16$, and the w-degree is 2. The horizontal axis of the figure corresponds to different CAPs, where the values 1, 2, 3, 4, and 5 correspond to DISTRIBUTE, PARTITION, RANDOM-SAME, SHIFTED, and SHUFFLE, respectively. The figure also has blocking curves for DFS when the wavelength preferences are Pack and Fixed.

Notice that shortest path channel assignment can depend on the exact value of $a$. However, it is not very sensitive with $a$. Also notice that the performance of shortest path assignment is more robust than DFS for different CAPs. However, for those CAPs with lowest blocking probabilities (DISTRIBUTE and SHIFTED), DFS with Fixed performs best, though only slightly better than shortest path.

Figure 4.8 presents the same type of blocking curves for the ring except that the w-degree is 4. The same conclusions can be drawn.

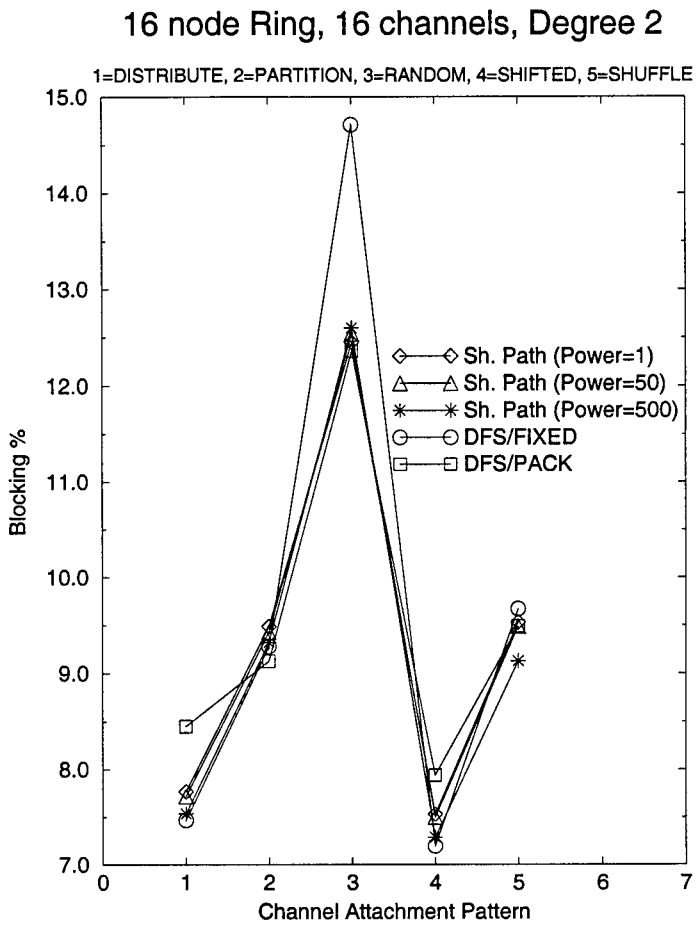



Figure 4.7: Blocking probabilities for the ring and w-degree 2.

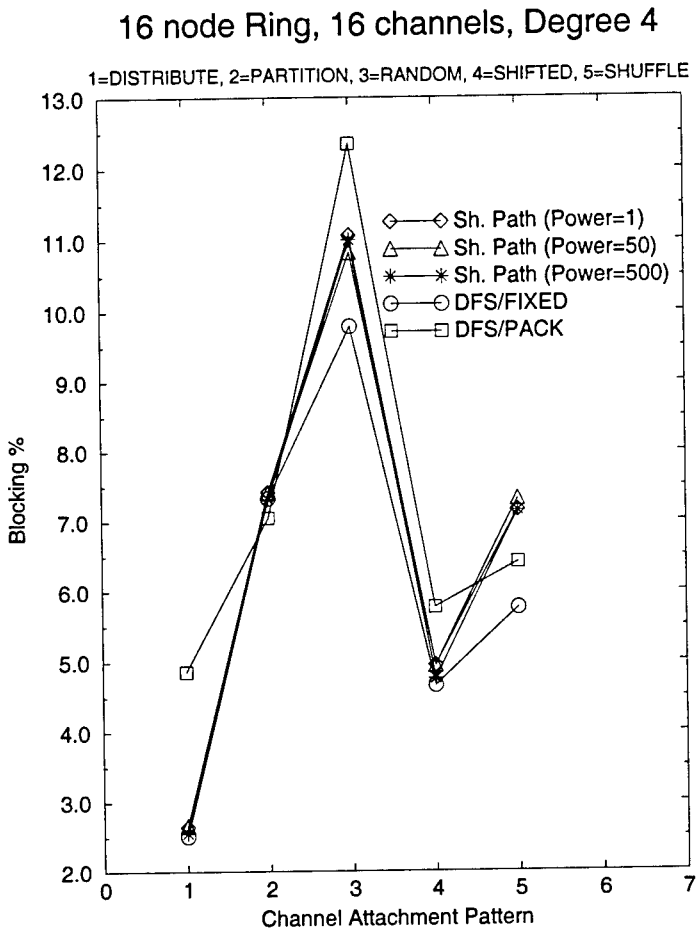



Figure 4.8: Blocking probabilities for the ring and w-degree 4.

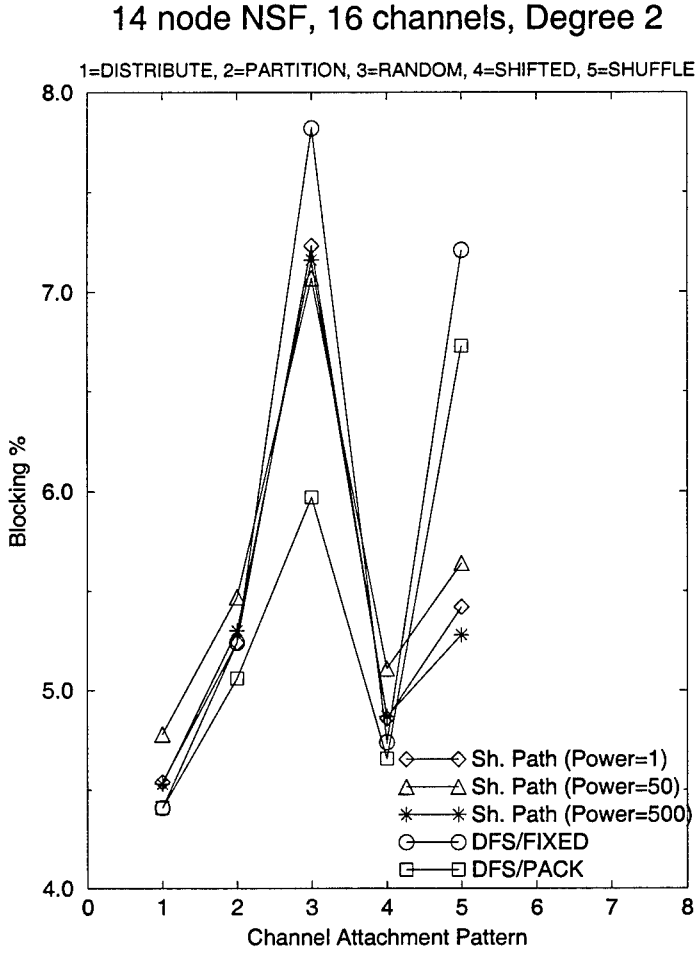



Figure 4.9: Blocking probabilities for NSFNET and w-degree 2.

Figure 4.9 has blocking curves for NSFNET and w-degree 2. For the CAPs that lead to low blocking probabilities (i.e., DISTRIBUTE and SHIFTED), DFS has better performance. However, the performance improvement is small.

From the simulations, we can conclude that shortest path can improve on DFS. However, it depends on the CAP and topology. For CAPs that lead to low blocking probabilities, shortest path and depth first search have small differences in blocking probability.

### 4.1.2 Channel Assignment Patterns (CAPs)

We will describe the simulation results for CAPs as follows.

- We will compare the performances of CAPs. For the channel assignment, we will use DFS with Fixed and Pack since this channel assignment leads to low blocking, as shown in the previous subsection. It will be shown that for a given w-degree

local CAPs (DISTRIBUTE, PARTITION, and SHIFTED) perform the best.

- We will show how the performance of the CAPs change with increasing w-degree. The CAPs considered are DISTRIBUTE and PARTITION, which were shown to minimize blocking.

Before we discuss the results, we should note that performance for SHIFTED was not optimized. In other words, the determination of which nodes should have the PARTITION CAP and which should have the S-PARTITION CAP was done arbitrarily rather than to optimize performance. In particular, nodes that were numbered with an odd value had the S-PARTITION, while nodes that were even valued had the CAP PARTITION.

We will now present our simulation results that compare the performances of different CAPs. Figure 4.10 shows the blocking probability curves for different CAPs for the ring topology, w-degree 2, and DFS with Fixed as a function of the number of wavelengths $W$. For comparison, the blocking curve for the case of no conversion and DFS with Fixed is included. The figure shows that local CAPs work better than expanding ones. Of the local CAPs, DISTRIBUTE and SHIFTED work best and have similar performance. Expanding CAPs have poor blocking probabilities, even exceeding the blocking probabilities of the no conversion case.

Figure 4.11 presents the blocking probability curves for different CAPs for the ring topology, w-degree 2, but for Pack rather than Fixed. The results are the same except that the differences in blocking probabilities are smaller.

Figures 4.12 and 4.13 compares CAPS for the NSFNET topology, w-degree 2, and for Fixed and Pack, respectively. The results are the same as for the ring. However, the expanding CAPs work better when compared to no conversion, especially, SHUFFLE and RANDOM-SAME.

We also have simulation results comparing the CAPs for the ring and NSFNET for w-degree 4, and for other topologies: trees, stars, mesh-connected rings, and randomly generated topologies. These are presented in Appendix B. The comparisons show that local CAPs are better than expanding ones, and DISTRIBUTE and SHIFTED work the best.

For the remainder of this section, we show how the performance of the network improves with increasing w-degree. We will consider local CAPs since they lead to better

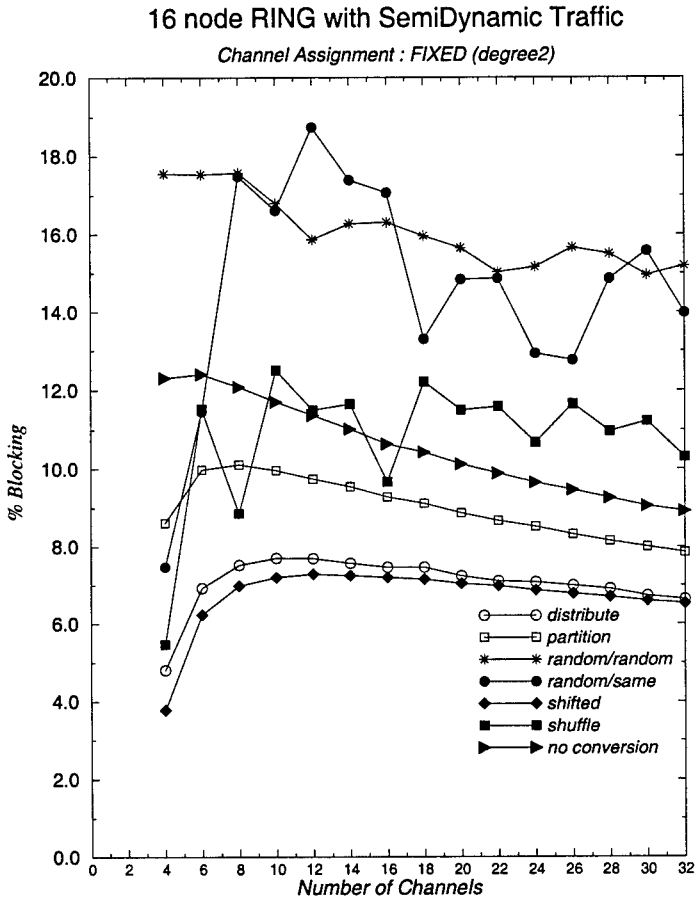



Figure 4.10: Blocking probabilities for the 16 node ring, w-degree 2, and DFS with Fixed.

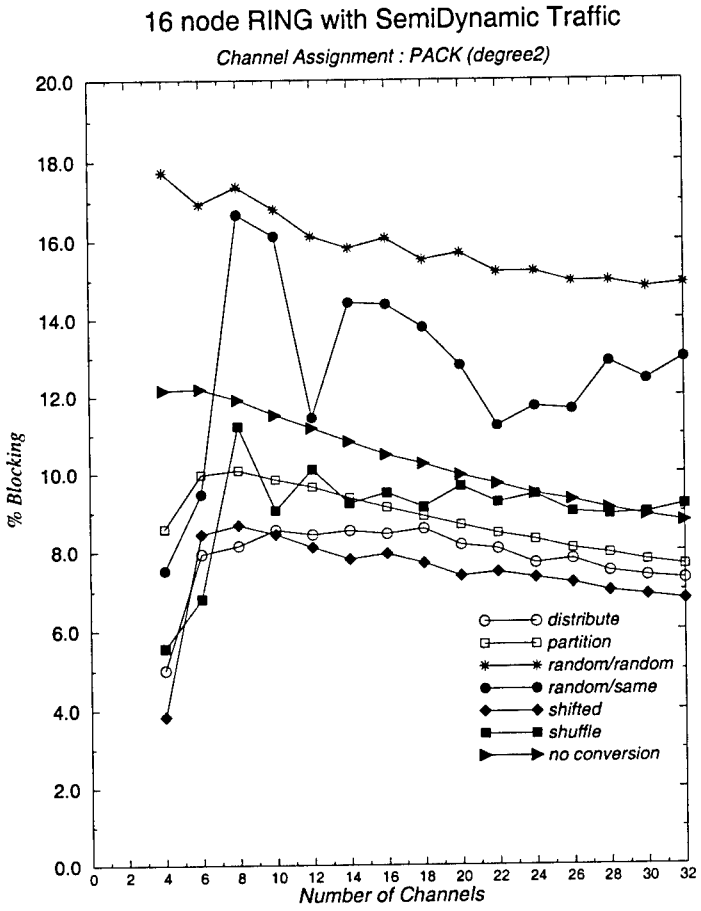



Figure 4.11: Blocking probabilities for the 16 node ring, w-degree 2, and DFS with Pack.

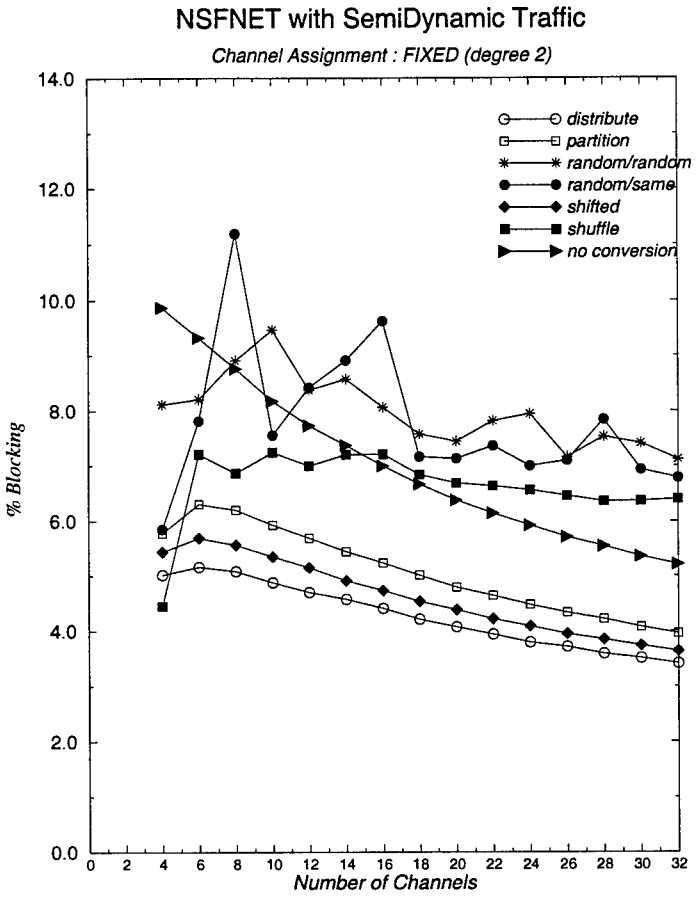



Figure 4.12: Blocking probabilities NSFNET, w-degree 2, and Fixed.

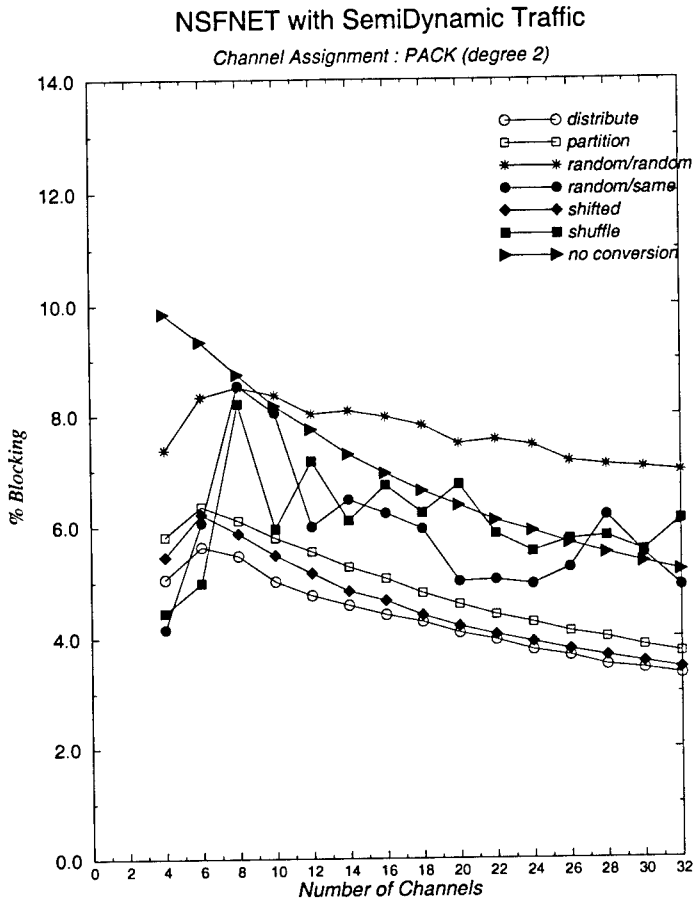



Figure 4.13: Blocking probabilities for NSFNET, w-degree 2, and Pack.

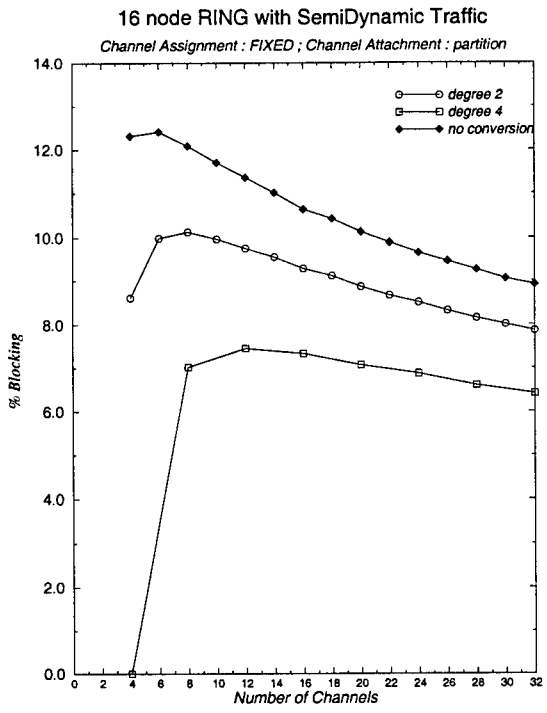



Figure 4.14: Blocking probabilities for the 16 node ring, Fixed, and PARTITION.

performance. In addition, we compare PARTITION and DISTRIBUTE since they are the extremes of the local CAPs, where one does not allow lightpaths to cross wavelength boundaries while the other does. For the channel assignment algorithm, we use DFS with Fixed.

Figure 4.14 shows the blocking probabilities for the 16 node ring and PARTITION for w-degrees 1 (no conversion), 2, and 4. The blocking probabilities decrease with larger w-degree. Figure 4.15 shows the blocking probabilities for the ring and DISTRIBUTE. The blocking probabilities decrease with larger w-degree, but the decrease is more significant. For example, consider $W = 32$. Then no conversion results in a blocking of 9%. The blocking probability will decrease to a litte more than 6% if the PARTITION CAP is used and w-degree is 4 (see Figure 4.14). However, for the same w-degree, but using the DISTRIBUTE CAP, the blocking goes down to around 3% (see Figure 4.15). This decrease to 3% from 9% illustrates that a significant improvement results from a moderate increase in w-degree.

Figures 4.16 and 4.17 shows the blocking probabilities for PARTITION and DISTRIBUTE, respectively, for NSFNET. Again the blocking decreased with larger w-degree, and DISTRIBUTE leads to smaller blocking.

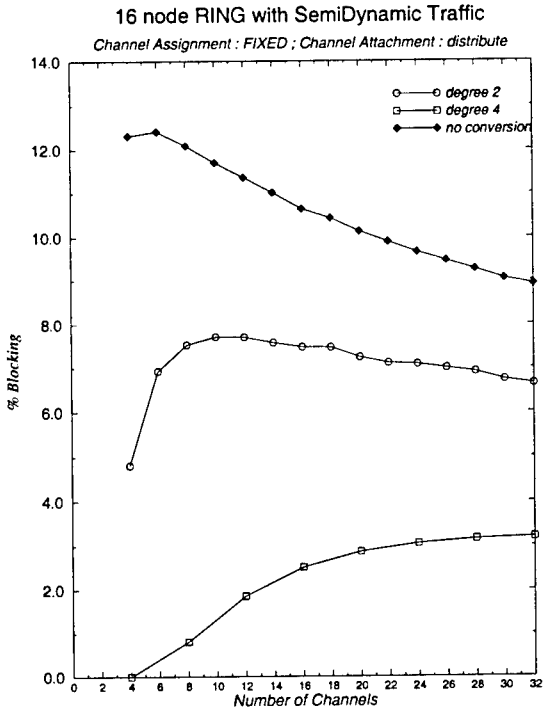



Figure 4.15: Blocking probabilities for the 16 node ring, Fixed, and DISTRIBUTE.

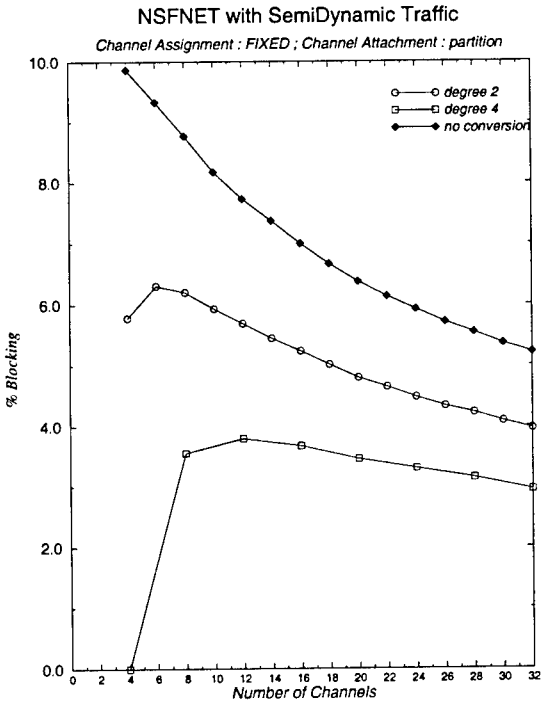



Figure 4.16: Blocking probabilities for NSFNET, Fixed, and PARTITION.

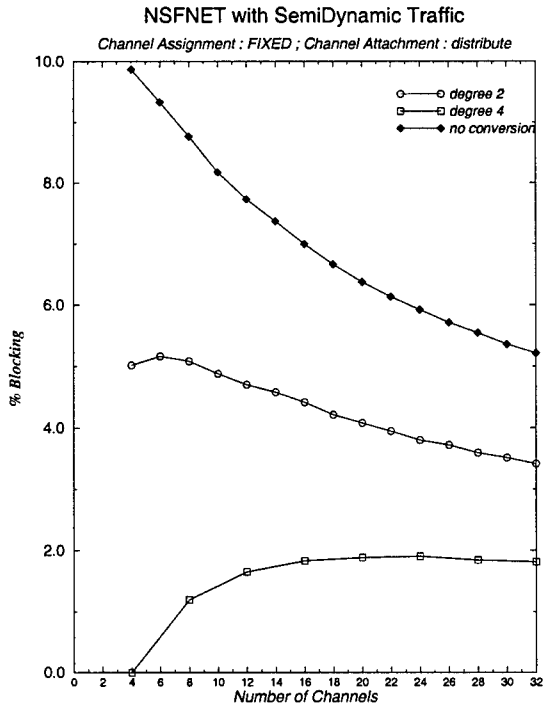



Figure 4.17: Blocking probabilities for NSFNET, Fixed, and DISTRIBUTE.

### 4.1.3 Channel Assignment With Rearrangement

We will compare the following types of channel assignment algorithms.

- No rearrangement. This is the DFS channel assignment algorithm with Fixed wavelength priorities. Recall that this algorithm does not rearrange existing lightpaths.
- Rearrangement after shut down (RAS).
- Single-MTV. This is a Move-To-Vacant algorithm where only one lightpath is moved.
- Multiple-MTV. This is a Move-To-Vacant, where multiple lightpaths may be moved.

To compare the algorithms we ran simulations. The network topologies considered were the 16 node ring and NSFNET. In addition, for most of the simulations we used the DISTRIBUTE CAP, since this CAP was shown to perform well as discussed in the previous subsection.

Figure 4.18 shows the various blocking percentages for the ring network with 16 wavelengths. The horizontal axis corresponds to CAPs with different w-degrees. In

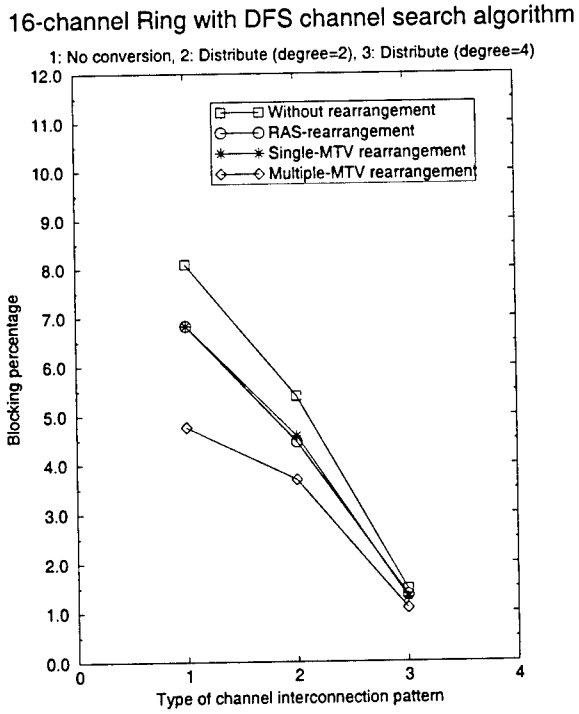



Figure 4.18: Comparison of Channel Assignment with Rearrangement for the Ring with 16 wavelengths

particular, they are no conversion, DISTRIBUTE with w-degree 2, and DISTRIBUTE with w-degree 4. The curves correspond to different channel assignment algorithms. Note that RAS, Single-MTV, and Multiple-MTV are all improvements over no-rearrangement. The improvement is less as the w-degree increases. Compared to no-rearrangement, the average reduction in blocking percentages obtained by using RAS, Single-MTV and Multiple-MTV are 15.44%, 15.03% and 6.25% respectively. Here, the "average" is over the three CAPs. The average blocking percentages of RAS, Single-MTV, and Multiple-MTV are 4.21%, 4.23% and 3.18% respectively.

Note that RAS and Single-MTV perform about the same. This makes a case for MTV since it is less disruptive to service but apparently without any significant degradation in blocking probabilities. Multiple-MTV performs the best, and the reduction in blocking probability can be signficant, especially in the case of no conversion or w-degree 2.

Figure 4.19 shows the various blocking percentages for the ring network with 32 wavelengths. The performance is similar. RAS and Single-MTV perform about the same, and Multiple-MTV performs best. We note that, compared to the no-rearrangement case, the reduction in average blocking percentage due to the RAS, Single-MTV and

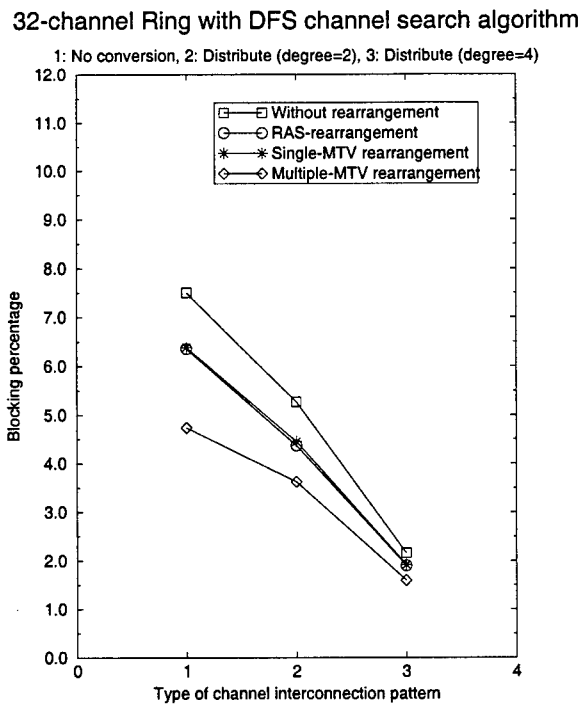



Figure 4.19: Comparison of Channel Assignment with Rearrangement for the Ring with 32 wavelengths

Multiple-MTV are 15.38%, 14.57% and 33.31% respectively. Multiple-MTV performs the best with an average blocking percentage of 3.32%, followed by RAS (4.21%) and Single-MTV (4.25%).

Figures 4.20 and 4.21 shows the various blocking percentages for the NSFNET network with 16 and 32 wavelengths, respectively. Again we have similar results except the reduction in blocking from no rearrangement is larger. Compared to the no-rearrangement case, the reduction in average blocking percentage obtained by using RAS, Single-MTV and Multiple-MTV is 38.03%, 36.10% and 60.07% respectively for the case of 16 wavelengths. For 32 wavelengths, the average reduction in blocking percentage for RAS, Single-MTV, and Multiple-MTV is 37.63%, 36.76% and 65.45% respectively.

From the above four experiments, we conclude that any rearrangement strategy (RAS, single-MTV, multiple-MTV) always leads to a reduction in blocking percentage when compared to the no-rearrangement case. The RAS and single-MTV rearrangement techniques perform in a very similar fashion for all the cases. Thus, given a choice between RAS and single-MTV, we always choose the latter because it never requires shutdown of lightpaths. For both mesh and ring topologies, it is seen that multiple-MTV performs

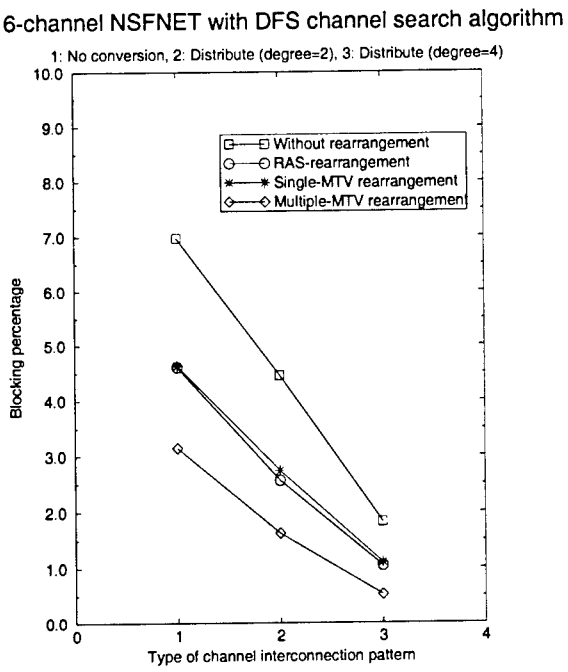



Figure 4.20: Comparison of Channel Assignment with Rearrangement for the NSFNET with 16 wavelengths

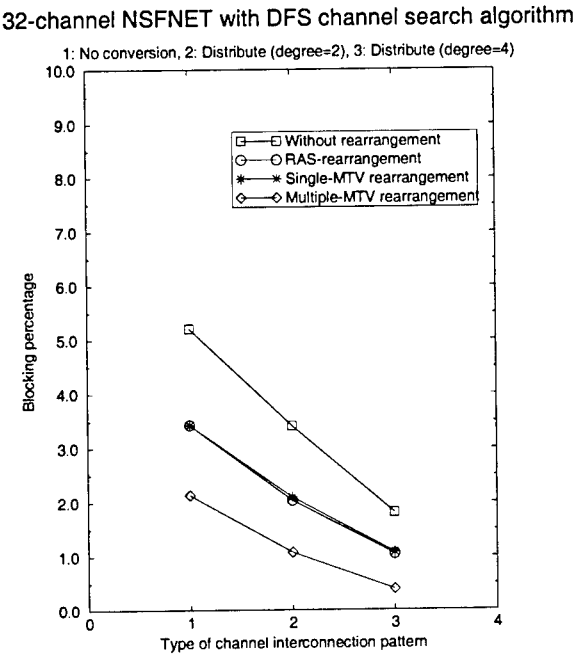



Figure 4.21: Comparison of Channel Assignment with Rearrangement for the NSFNET with 32 wavelengths

better than single-MTV.

Finally, we compare our results with those obtained by [21]. In their paper, they consider the improvement in performance obtained by using multiple-MTV rearrangement in no conversion networks. They use the *dynamic* traffic model in which lightpaths arrive as a Poisson processes and have exponential holding times. They show that using multiple-MTV leads to a significant improvement in blocking probability in no conversion networks. In fact, the performance curve for no-conversion with multiple-MTV is close to that of full conversion. We have generalized their multiple-MTV algorithm for limited conversion networks. Even in our experiments, use of multiple-MTV in limited conversion networks led to a significant reduction in blocking percentage. In some cases, the blocking percentage using multiple-MTV was very close to zero. Thus, our results are analogous to those obtained by [21].

### 4.1.4 Channel Assignment Without Pre-Computed Routes

In this subsection, we will discuss our simulation results for channel assignment algorithms without pre-computed routes. Recall that these algorithms are

- *Widest Shortest.* This algorithm finds a channel assignment in two steps. First, it finds a *widest shortest* route based on the free bandwidth of the fiber links. Then it finds a channel assignment along the route. In our experiments, the channel assignment used was DFS with Fixed wavelength priorities.

- *Shortest Distance.* This algorithm finds the shortest distance channel assignment, where each channel has *distance* value $1/r_e$, $r_e$ is the amount of free channels in the link $e$, and link $e$ is the link for the channel.

- *Brute Force Search.* This is the modified *breadth first search* algorithm, which explores channels in groups, that depend on their hop-distance from the source. The order that the channels within a group are explored is dependent on a wavelength preference. In our experiments we used Fixed.

We have the following set of results.

- We will compare the performances of the channel assignment algorithms. The CAP will be DISTRIBUTE. The results are for REALNET (Figure 3.3) and a couple of other topologies. We will show that Shortest Distance is best.

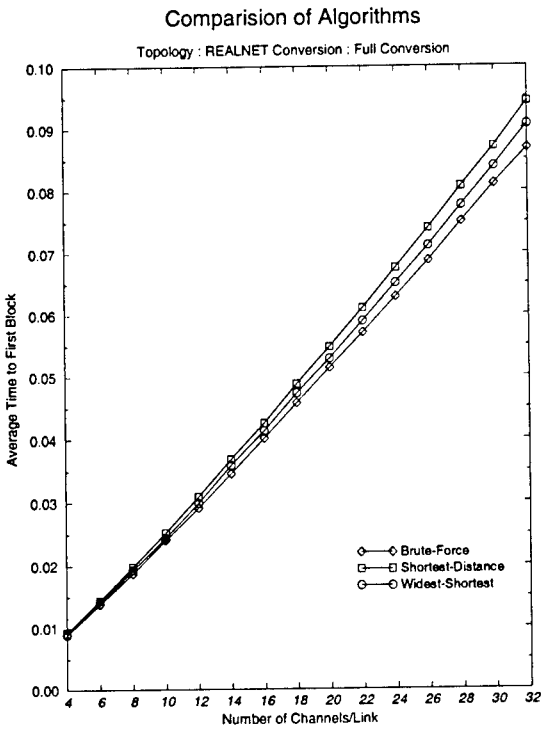



Figure 4.22: Comparison of Algorithms — REALNET Topology with Full Conversion

- We will show how the performance changes with w-degree. The CAPs used were DISTRIBUTE and PARTITION.
- We compare the local CAPs DISTRIBUTE, PARTITION, and SHIFTED using Shortest Distance. It will be shown that DISTRIBUTE works best. We will also show that SHIFTED can be better than PARTITION even though these two CAPs are similar.

We will first compare the channel assignment algorithms on the REALNET topology. Figure 4.22 shows the average time to first block in a REALNET with full conversion. The simulations were run for different numbers of wavelengths, as show on the horizontal axis. As the number of wavelengths increases, the average time to block increases since it then takes more time to fill up the network. For this full conversion case, the performances are about the same, but Shortest Distance performs best, Widest Shortest is the next best, and Brute Force Search is the worst.

Figures 4.23 and 4.24 shows the average time to first block for the algorithms when REALNET has no conversion and limited wavelength conversion, respectively. For the limited wavelength conversion network, the CAP was DISTRIBUTE with w-degree 4.

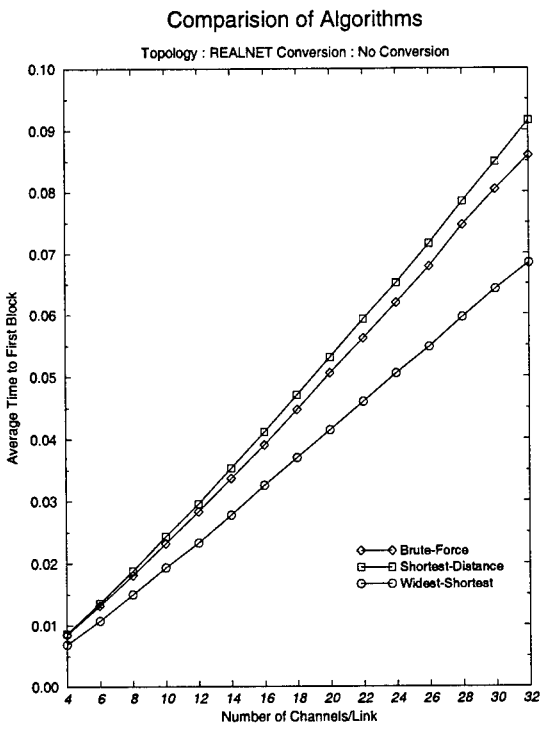



Figure 4.23: Comparison of Algorithms — REALNET Topology with No Conversion

Shortest Distance works best in both cases. However, now Brute Force Search is next best followed by Widest Shortest. The reason why Widest Shortest performs poorly is in the way it finds routes for lightpaths. It computes them by using the number of free channels in each link and does not consider the CAPs at each node. Therefore, it can find a route that may have available free channels on each link but the channels may not be w-attached at intermediate nodes. On the other hand, Shortest Distance and Brute Force Search use the CAPs to determine channel assignments.

Figures 4.25 and 4.26 compare the algorithms for two other topologies, RANDOM (Figure 3.5) and NSFNET (Figure 3.2), respectively. For the RANDOM topology, the network had no conversion, and for the NSFNET topology, the network had full conversion. Shortest Distance performed best for both topologies. As expected, in the case of the no conversion network (RANDOM topology), Brute Force performed better than Widest Shortest, and in the case of full conversion (NSFNET), the opposite was true.

Next, we show how w-degree can affect the performance of the network using the channel assignment algorithms. The next set of figures show our simulation results. For these simulations, we used the DISTRIBUTE CAP. We also used the WORST-CASE topology. This topology will force lightpaths to have 7 or more hops and have

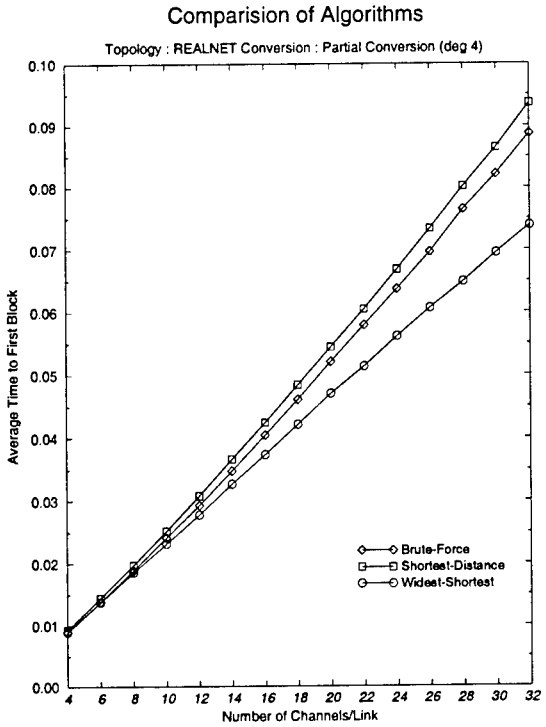



Figure 4.24: Comparison of Algorithms — REALNET Topology with DISTRIBUTE (w-degree 4)

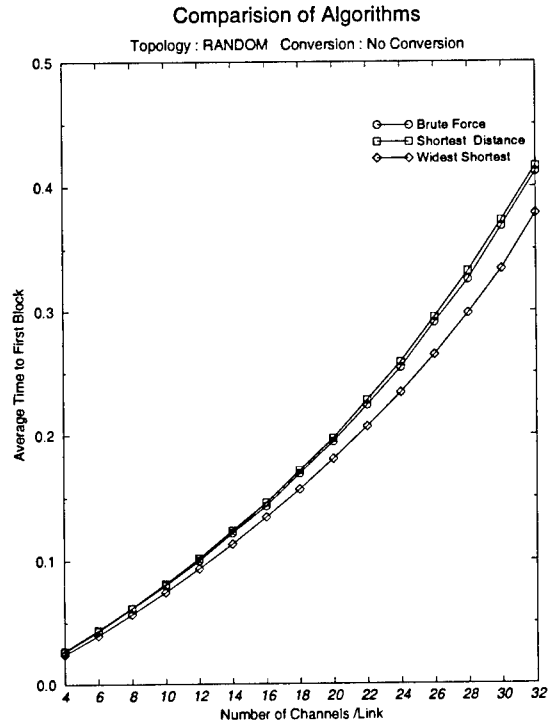



Figure 4.25: Comparison of Algorithms — RANDOM Topology with No Conversion

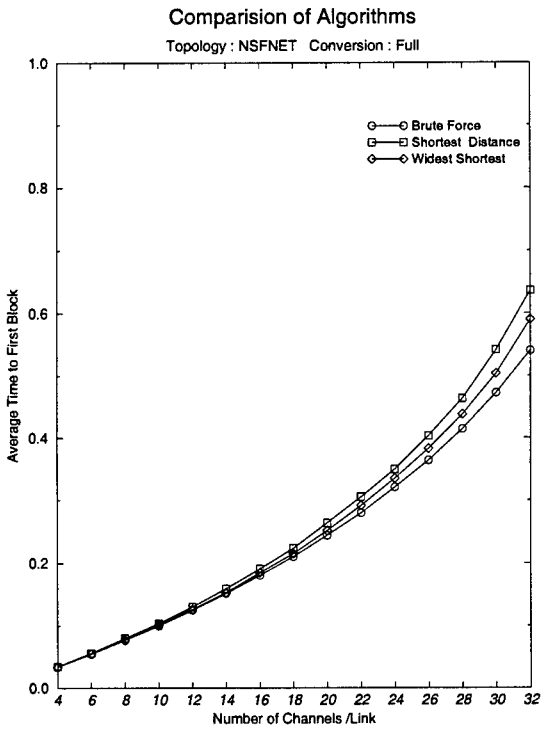



Figure 4.26: Comparison of Algorithms — NSFNET Topology with Full Conversion

them intersect at multiple links. Then wavelength conversion becomes important and we should see more of a difference in performance between no and full conversion.

Figure 4.27 shows the the performance for Brute Force Search for no conversion, w-degree 4, and full conversion. As expected the performance improves as the amount of conversion increases. In fact, w-degree 4 performs almost the same as full conversion. Also notice that there is a significant improvement in the average time until first block from no conversion to w-degree 4. The improvement is more than 40% in a network with 32 wavelengths. To illustrate that the WORST-CASE topology needs more wavelength conversion than other topologies Figure 4.28 shows the performance for Brute Force Search for the RANDOM topology. The random topology will have lightpaths that are shorter on average, and so wavelength conversion does not affect its performance as much.

Figures 4.29 and 4.30 show the results for Widest Shortest and Shortest Distance, respectively, for the WORST-CASE topology. When the number of wavelengths is 32, wavelength conversion can significantly improve the average time to first block. For Widest Shortest (Figure 4.29), limited wavelength conversion with w-degree 4 is significantly better than no conversion. For Shortest Distance (Figure 4.30), limited wavelength

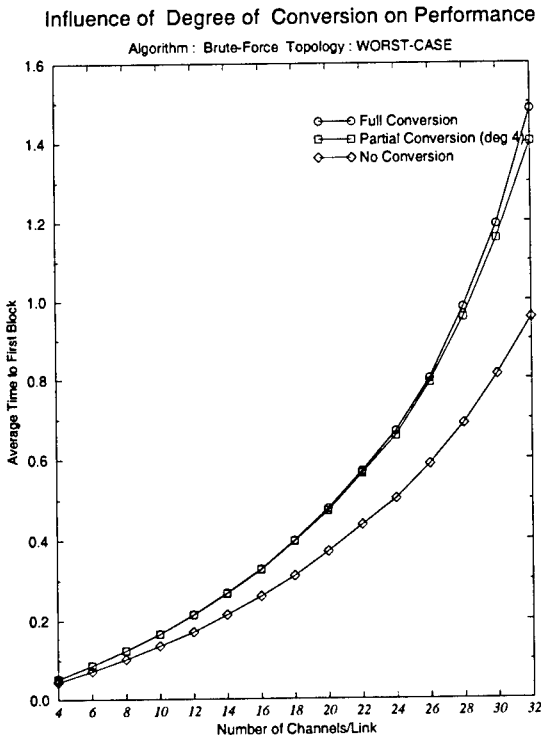



Figure 4.27: Plot for Brute Force Algorithm — WORST-CASE Topology, DISTRIBUTE CAP.

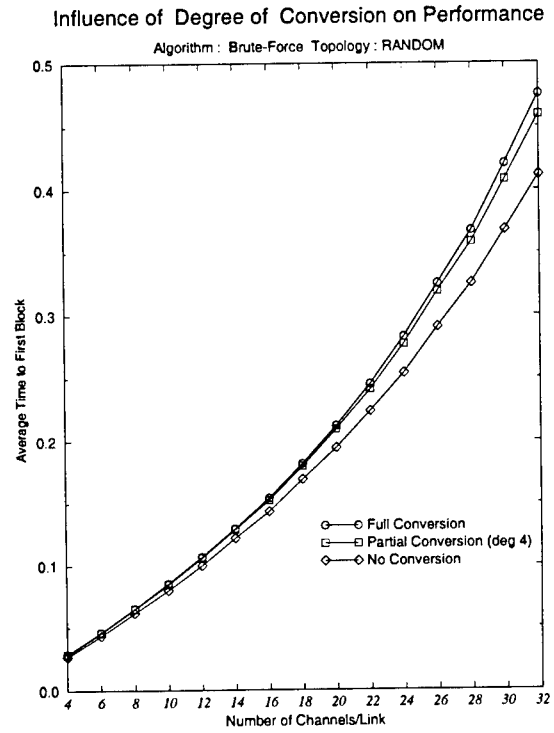



Figure 4.28: Plot for Brute Force Algorithm — RANDOM Topology, DISTRIBUTE CAP.

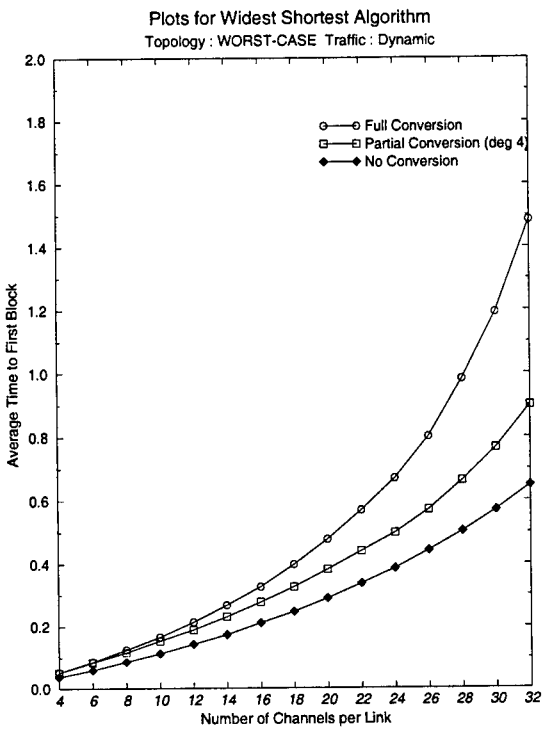



Figure 4.29: Performance variation with w-degree — Widest Shortest Algorithm on WORST-CASE Topology, DISTRIBUTE CAP.

conversion with w-degree 4 is the same as full conversion.

We also ran simulations to test the algorithms for the PARTITION CAP. Figure 4.31 present the results for the WORST-CASE topology and w-degree 4. Shortest Distance and Brute Force Search perform about the same and better than Widest Shortest. This is especially true at 32 wavelengths. Thus, qualitatively the results are similar as before.

Figures 4.32, 4.33, and 4.34 show our simulation results for Brute Force Search, Shortest Distance, and Widest Shortest, respectively, for the WORST-CASE topology and different w-degrees. Performance improves with increasing w-degree as expected. However, the improvement is more gradual. For w-degree 4, the PARTITION CAP does not perform as well as full conversion, though for the DISTRIBUTE CAP, it does.

Finally, we will compare the performances of the local CAPs DISTRIBUTE, PARTITION, and SHIFTED using the channel assignment Shortest Distance. The purpose of this comparison is to determine if there are any differences in performance between these local CAPs. We focused on local CAPs since they performed best as shown in Subsection 4.1.2. We limited ourselves to the Shortest Distance algorithm since it performed better

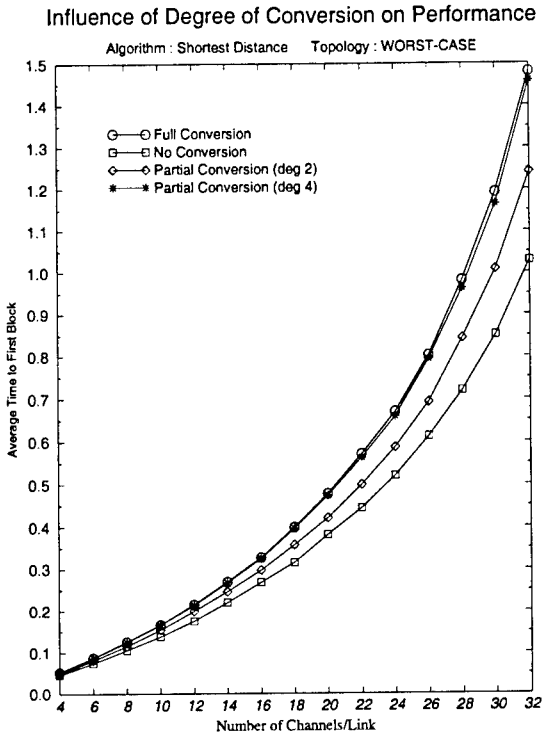



Figure 4.30: Performance variation with w-degree — Shortest Distance Algorithm on WORST-CASE Topology, DISTRIBUTE CAP.

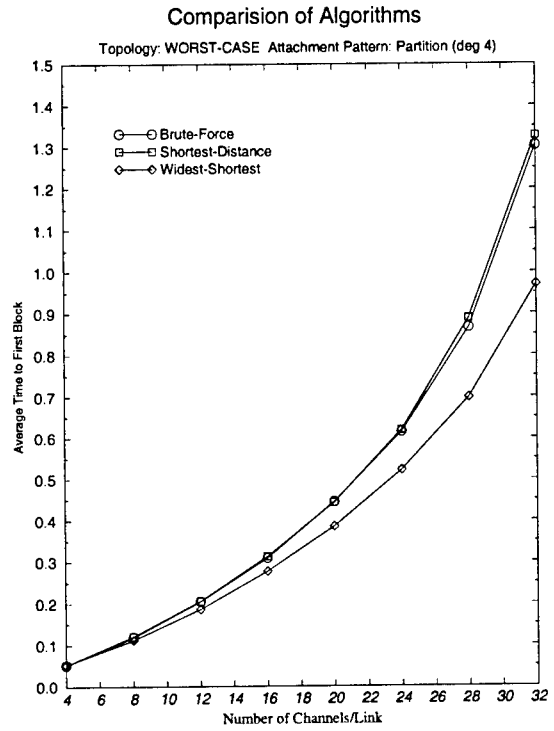



Figure 4.31: Comparison of Algorithms — WORST-CASE topology with PARTITION CAP and w-degree 4

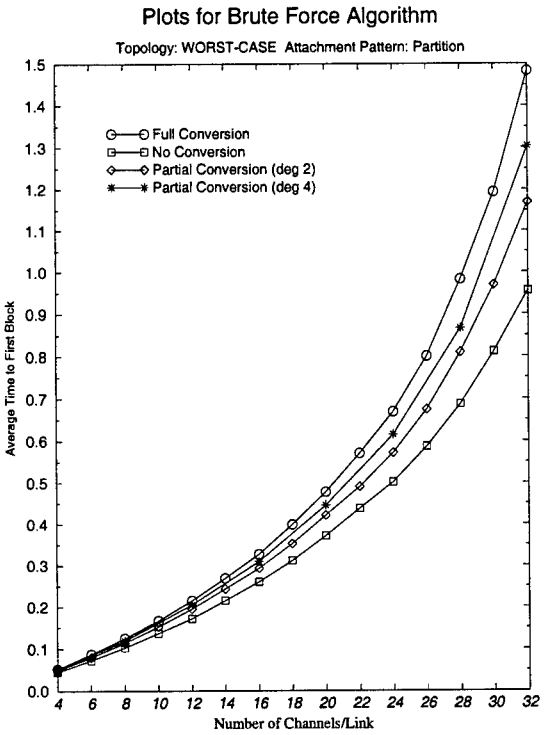



Figure 4.32: Performance variation with degree of conversion — Brute Force Algorithm on WORST-CASE Topology with PARTITION CAP.

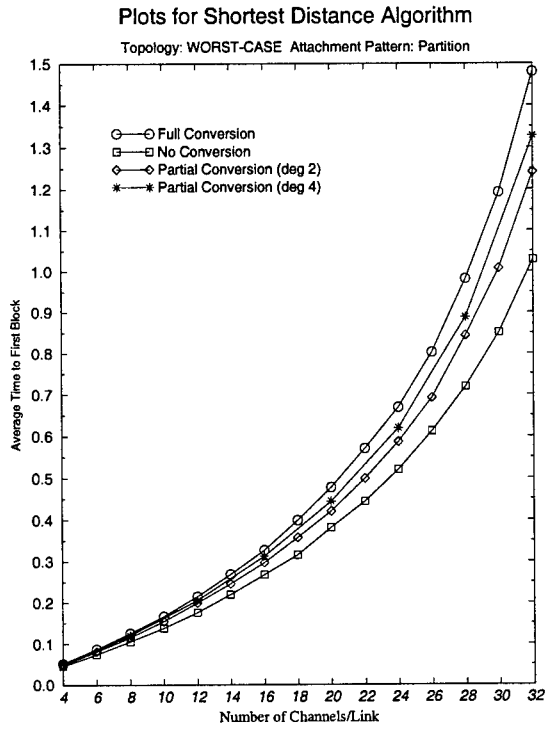



Figure 4.33: Performance variation with degree of conversion — Shortest Distance Algorithm on WORST-CASE Topology with PARTITION CAP.

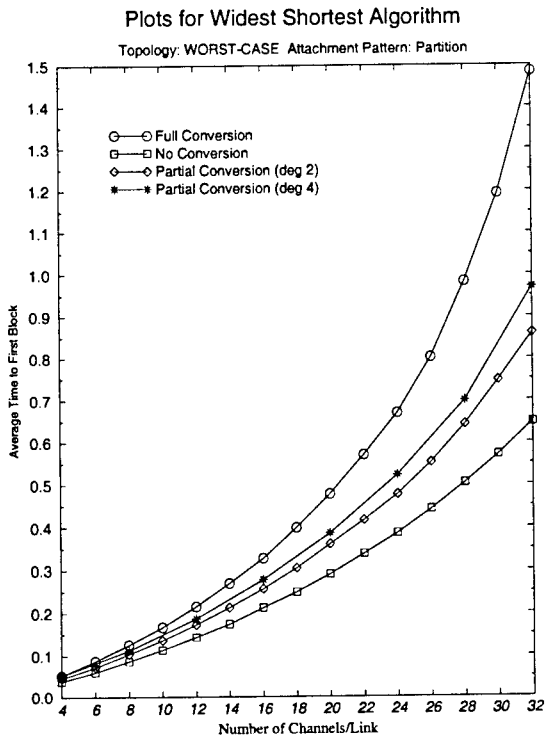



Figure 4.34: Performance variation with degree of conversion — Widest Shortest Algorithm on WORST-CASE Topology with PARTITION CAP

than Brute Force Search and Widest Shortest. Figures 4.35 and 4.36 show the results for w-degree 2 and 4, respectively. The performances are close, but DISTRIBUTE performs the best.

### 4.1.5 DesignCAP

In this subsection we present the performance results of the DesignCAP algorithm. The drawback of this algorithm is its prohibitive run time. To evaluate the cost function $f(c)$ of each CAP $c$, requires a simulation. One iteration of the algorithm requires computing $f(c')$ for all possible branch exchanges from the current CAP $c$. Therefore, we limited our results to a moderate number of wavelengths $W = 8$ and w-degree 2.

Figure 4.37 shows the results of our simulations. DesignCAP was run five times from five different starting points. These starting points are DISTRIBUTE' (a modification of DISTRIBUTE), SHUFFLE, RANDOM-1, RANDOM-2, and RANDOM-3. All have w-degree 2 and in fact each have their outgoing wavelengths w-attached to exactly 2 incoming wavelengths, and each of their incoming wavelengths w-attached to exactly 2 outgoing wavelengths. We will refer to such CAPs as *regular*. RANDOM-1,

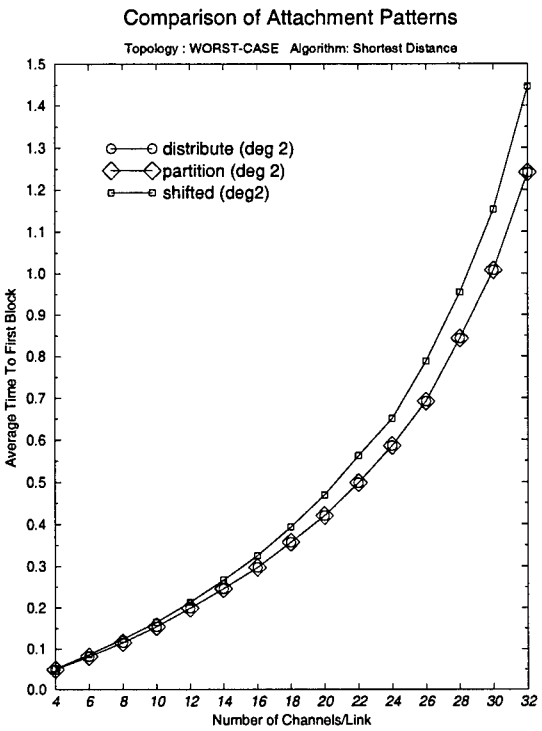



Figure 4.35: Comparison of Attachment Patterns — WORST-CASE topology and Shortest Distance Algorithm

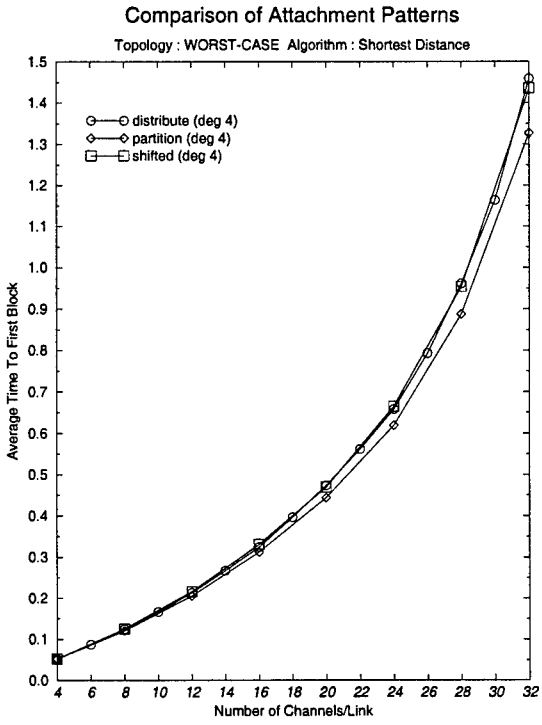



Figure 4.36: Comparison of Attachment Patterns — WORST-CASE topology and Shortest Distance Algorithm

| Starting CAP | # Iterations | Starting Cost | Final Cost |
|---|---|---|---|
| DISTRIBUTE' | 4 | 1.16 | 1.14 |
| SHUFFLE | 8 | 1.41 | 1.18 |
| RANDOM-1 | 5 | 3.19 | 1.47 |
| RANDOM-2 | 8 | 2.96 | 1.25 |
| RANDOM-3 | 11 | 3.04 | 1.20 |

Figure 4.37: Results of DesignCAP from different starting CAPs.

RANDOM-2, and RANDOM-3 were randomly chosen from among all possible regular CAPs. DISTRIBUTE' is a modification of DISTRIBUTE and it is used rather than DISTRIBUTE because DISTRIBUTE is not regular. DISTRIBUTE has an incoming wavelength ($w_7$) that is w-attached to *only one* outgoing wavelength. It also has an outgoing wavelength ($w_0$) that is w-attached to *only one* incoming wavelength. Thus, to be regular, DISTRIBUTE' has incoming wavelength $w_7$ w-attached to outgoing wavelength $w_0$.

Figure 4.37 shows for each of these starting CAPs, the number of iterations of DesignCAP. It shows the costs of the initial and final CAPs. (Recall that the cost is the average number of lightpath requests blocked in a batch of lightpath requests.) Notice that DISTRIBUTE' leads to the best final solution. Also notice that DesignCAP found significantly better CAPs from the initial ones except for DISTRIBUTE'. However, we suspect that DISTRIBUTE' is already nearly optimal. Figure 4.38 shows the CAPs found by DesignCAP over the four iterations starting from DISTRIBUTE'. Notice that only two distinct CAPs were found: DISTRIBUTE' and a slight modification. The algorithm went through additional iterations because of the randomness of the cost function $f(c)$ of the CAPs. Thus, the costs between these CAPs are about the same, and DISTRIBUTE' is in a local minimum. This gives evidence that the local CAPs such as DISTRIBUTE are nearly optimal.

As another example, Figure 4.39 shows SHUFFLE and the resulting final CAP computed by DesignCAP. We believe the algorithm tries to place w-attachments so that high preference wavelengths are attached to each other. For example, incoming wavelength $w_3$ was originally w-attached to outgoing wavelengths $w_6$ and $w_7$. However, in the final CAP, it is w-attached to $w_2$ and $w_3$. This may help to make the channel assignment

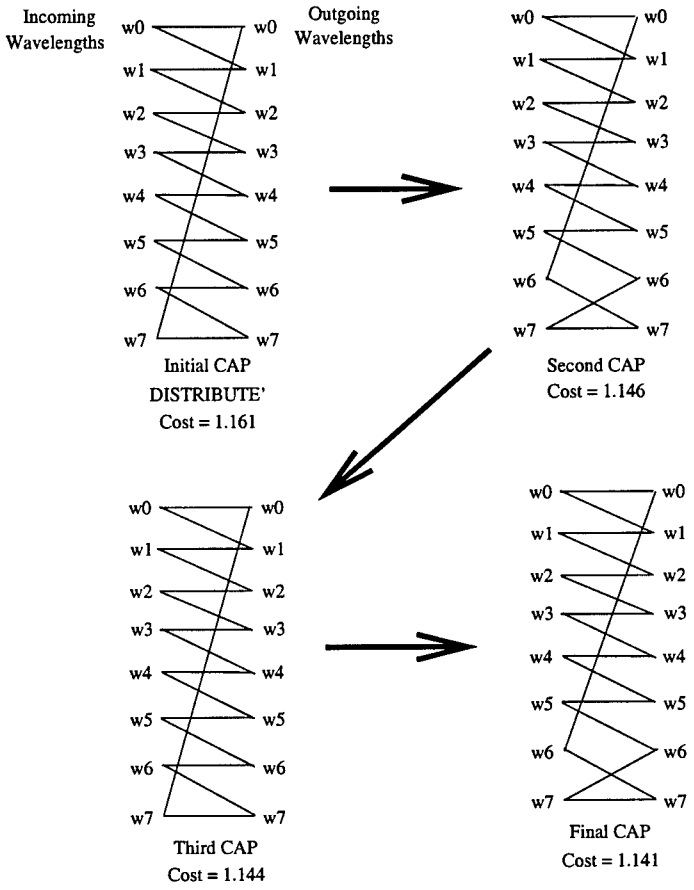



Figure 4.38: The CAPs found by DesignCAP starting from DISTRIBUTE'.

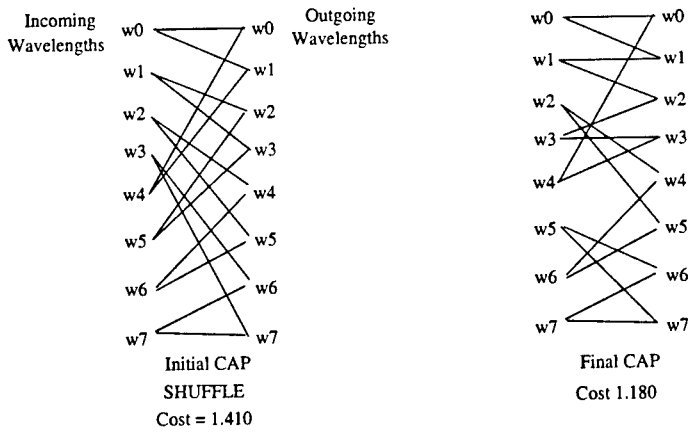



Figure 4.39: The CAP SHUFFLE and the final CAP found by DesignCAP starting from SHUFFLE.

algorithm, DFS with Fixed, to be more efficient since DFS with Fixed tends to pack lightpaths at wavelengths with smaller indices.

## 4.2 Networks with Minimal ADMs/DCSs

We will present our results for networks at the tributary traffic layer.

- **Rings**
  - *Nonblocking ring networks.* We first focus on ring topologies. We present a number of nonblocking networks.
  - *Design algorithm for Incremental ring networks.* We consider one of the nonblocking networks called an *Incremental* ring. We have a design algorithm for it which minimizes cost. We present simulations that shows the quality of solutions of the design algorithm.
  - *Simulations of random traffic for Incremental ring networks.* Incremental ring networks are nonblocking for *semi-dynamic* (or *incremental*) traffic, i.e., traffic never terminates. However, they are only rearrangeably nonblocking for dynamic traffic that are allowed to terminate. In other words, there are cases when dynamic traffic will lead to blocking unless existing traffic may be rearranged. We simulated the Incremental network for random traffic to determine how frequent these blocking cases arise.

The results of the ring networks are discussed in Subsection 4.2.1

- **General Topologies**
  - *A nonblocking general topology network.* We present a general topology network that is nonblocking for semi-dynamic traffic.
  - *Design algorithm for general topology networks.* We present a design algorithm for the general topology networks. We present simulations that shows the quality of the algorithm's solutions.
  - *Simulations of random traffic for general topology networks.* Though the network is nonblocking for semi-dynamic traffic, it is only rearrangeably blocking for dynamic traffic. There are cases when dynamic traffic will lead to blocking unless existing traffic can be rearranged. We simulated the network for random traffic to determine the blocking probabilities.

The results of the general topology networks are discussed in Subsection 4.2.2

### 4.2.1 Rings

#### Nonblocking Ring Networks

We found several nonblocking ring networks. These are described in detail in [15], where Appendix C is a copy.

- **Single-Hub:** This network has a unique node designated as the *hub*, which has lightpaths directly connecting it to all other nodes. It is wide sense nonblocking for dynamic traffic satisfying Assumption B.1. It has a small number of ADMs but a large number of wavelengths.
- **Double-Hub:** This network has two hubs, which have lightpaths connecting them to all other nodes. It is rearrangeably nonblocking for dynamic traffic satisfying Assumption B.1. It has small number of ADMs and wavelengths. Therefore, it is efficient in both the number of ADMs and wavelengths. However, it is rearrangeably nonblocking rather wide-sense nonblocking.
- **Hierarchical Ring:** This is a simple network composed of two point-to-point subrings overlayed on a common physical ring, and is wide-sense nonblocking for

dynamic traffic satisfying B.2. It has more wavelengths than a point-to-point network, but may use less ADMs.

- **Incremental Ring:** This is a ring network that is organized (recursively) from sections of the ring. It is wide-sense nonblocking for semi-dynamic traffic that satisfies Assumption B.2. It is also rearrangeably nonblocking for dynamic traffic that satisfies Assumption B.2. It uses the minimum number of wavelengths, so it uses the same number of wavelengths as a point-to-point network. However, it uses less ADMs.

For the rest of the study, we focused on the Incremental ring because it is as bandwidth efficient as a point-to-point network, yet reduces the number of ADMs.

The basic idea of why this network is nonblocking is illustrated in Figure 4.40. The left side of the figure shows a section of a point-to-point network with $W = 8$ wavelengths. This network is nonblocking for dynamic traffic satisfying Assumption B.2. The right hand side shows the same section but where $t(i) = 2$ ADMs have been dropped from node $i$. In order to insure that this right hand side is nonblocking for semi-dynamic traffic, we need to carefully define the assignment of the tributary traffic onto wavelengths. In particular, the wavelengths at node $i$ are designated as either *transit* or *local*. Local wavelengths are for traffic terminating at node $i$, but can also be used by transit traffic. At the node, the local wavelengths have ADMs, while the transit wavelengths have none. The assignment of traffic is such that local traffic is assigned to local wavelengths and transit traffic is assigned to transit wavelengths. Note that transit traffic may fill up transit wavelengths, and then they may use local wavelengths. The network remains wide sense nonblocking because transit traffic fills transit wavelengths first; and local traffic are not blocked because there are enough ADMs provisioned at local wavelengths.

This idea can be extended further. Instead of dropping ADMs at one node, we can drop ADMs over a section of the ring. Terminating traffic now corresponds to ones that terminate at some node in the section. Transit traffic corresponds to traffic that bypasses the section. A complete description of why the Incremental network is nonblocking for semi-dynamic traffic is given in Appendix C [15].

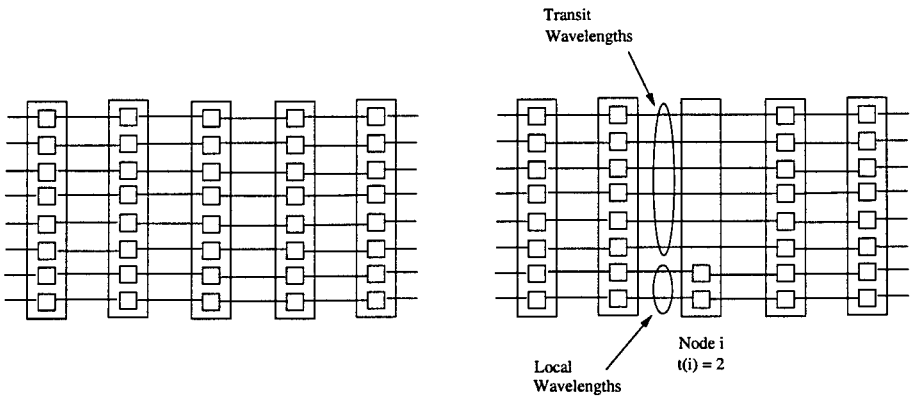



Figure 4.40: A section of a ring with $W = 8$. On the left is the section when is a part of a point-to-point network. On the right is when nodes are deleted from its transit wavelengths.

We should note that under Traffic Assumption B.3, the right hand side of Figure 4.40 is wide-sense nonblocking for dynamic traffic. Recall, that Assumption B.3 is Assumption B.2 plus the additional constraint that for each node $i$, the amount of transit traffic is at most $g \cdot (L - t(i))$. However, it appears that the wide-sense nonblocking property cannot be extended beyond this simple case. We considered the assumption because we thought it may lead to wide-sense nonblocking results for dynamic traffic.

### Design Algorithm for Incremental Rings

We developed a design algorithm for the Incremental ring to minimize its cost. This algorithm is described in detail in Appendix D [29]. It applies dynamic programming to get a solution.

To evaluate its solutions, we ran simulations over randomly generated values of $\{t(i)\}$ for $W = 32$ and rings of size 8, 12, and 16. Figure 4.41 shows the results of the solutions for different values of $F$, the parameter used to characterize the random $\{t(i)\}$ values. Recall that $\{t(i)\}$ are generated by creating random traffic that exactly fills up the $W$ wavelengths on every link (see Subsection 3.2.2). $F$ is a measure of variation between the $\{t(i)\}$ values. The larger $F$ is the more variable $\{t(i)\}$ is.

Each data point is the ratio of the average number of ADMs per node found by the algorithm over $W$ (recall that $W$ is also the number of ADMs per node for a point-to-point network). A hundred samples were generated to get a data point.

Notice that the number of ADMs per node gets smaller with increasing $N$ and $F$

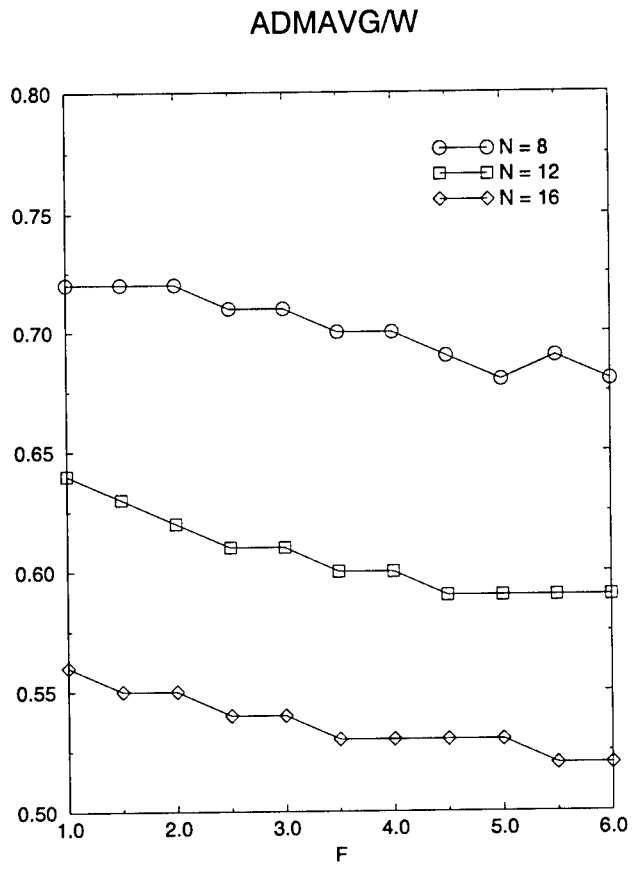



Figure 4.41: The ratio $ADMAVG/W$ as a function of $F$ for $N = 8, 12$, and 16.

($\{t(i)\}$ is more variable). Notice that the cost savings from using a point-to-point network is at least 27%, 36%, and 44% for $N = 8, 12$, and 16, respectively. This can be considerable. For example, if ADMs cost \$100,000 then a 27% savings amounts to saving \$864,00 per node, and a 44% savings amounts to a savings of \$1.4 million per node.

### Simulations of Random Traffic for Incremental Rings

The Incremental ring is wide-sense nonblocking for semi-dynamic traffic that satisfies Assumption B.2. However, it is only rearrangeably nonblocking for dynamic traffic that satisfies the assumption. To determine if it blocks typical dynamic traffic, we simulated it for the conforming Poisson traffic, described in Section 4.2. We simulated two networks.

- **Ring 1:** For this ring $N = 8$, $W = 8$, and $g = 4$. In addition, it was assumed that for all nodes $i$, $t(i) = 4$. At nodes $\{0, 2, 4, 6\}$, all 8 wavelengths were terminated by ADMs. At nodes $\{1, 3, 5, 7\}$, only four wavelengths $\{0, 1, 2, 3\}$ were terminated by ADMs. This placement of ADMs is for the uniform traffic pattern where each each pair of nodes will have on average $g = 4$ traffic streams between them.

  We ran simulations for different values of $a$, where $a$ corresponds to the aggregate Poisson traffic rate. Each simulation run would stop after 2 million conforming traffic streams. Figure 4.42 shows the average channel utilization for the ring for different values of $a$. In these simulations, no blocking was recorded. Notice that the utilization increased and then decreased as the arrival rate increased. Initially the utilization would increase as the network became more loaded. However, when the arrival rate became high, the distribution of the lengths of the conforming traffic streams changed. The lengths became shorter due the high link utilization, and because short routes were more easily conforming than long routes. Since traffic stream routes have shorter lengths, their contribution to the channel utilization became smaller.

  Table 4.43 shows the distribution of traffic stream lengths. Notice that for small $a$ the traffic streams are approximately uniformly distributed on $[1, N/2]$. However, as $a$ became larger, arriving traffic streams with long route lengths will tend to be less conforming. Thus, they are ignored. Hence, as $a$ gets larger, conforming traffic streams will be more likely to have short routes.

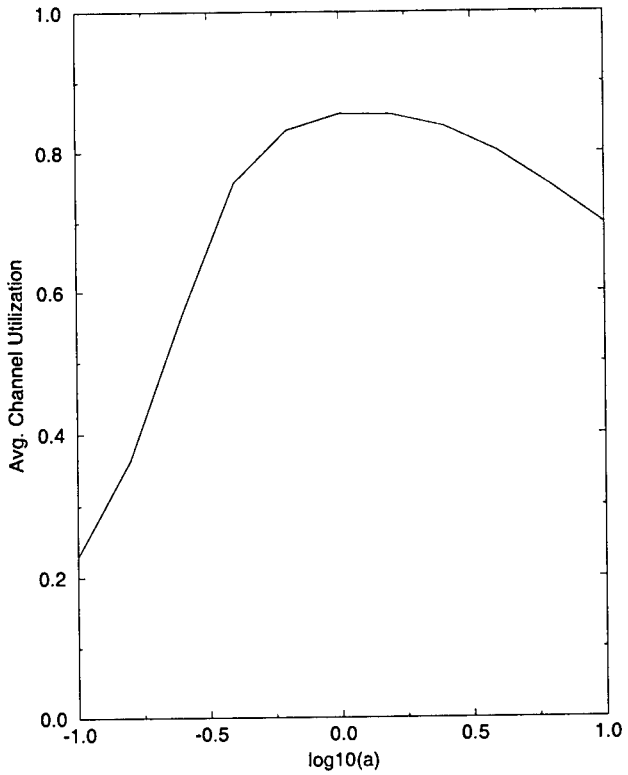



Figure 4.42: Average WDM channel utilization for a ring network with $N = 8$, $W = 8$, $g = 4$, and $t(i) = 4$ for all nodes $i$. The average channel utilization is a function of $log_{10}(a)$. For this plot, the value of $a$ ranges from 0.1 to 10.0.

| | Fraction of accepted streams with length $h$ | | | |
|---|---|---|---|---|
| $log_{10}(a)$ | $h = 1$ | $h = 2$ | $h = 3$ | $h = 4$ |
| -1.0 | 0.286 | 0.285 | 0.286 | 0.143 |
| -0.8 | 0.286 | 0.285 | 0.286 | 0.143 |
| -0.6 | 0.287 | 0.285 | 0.285 | 0.143 |
| -0.4 | 0.306 | 0.280 | 0.277 | 0.137 |
| -0.2 | 0.352 | 0.269 | 0.254 | 0.125 |
| 0.0 | 0.410 | 0.251 | 0.226 | 0.113 |
| 0.2 | 0.476 | 0.228 | 0.194 | 0.102 |
| 0.4 | 0.548 | 0.199 | 0.163 | 0.091 |
| 0.6 | 0.622 | 0.167 | 0.132 | 0.079 |
| 0.8 | 0.697 | 0.134 | 0.103 | 0.066 |
| 1.0 | 0.768 | 0.104 | 0.077 | 0.051 |

Figure 4.43: Distribution of the lengths of traffic streams that are accepted by the ring network with $N = 8$, $W = 8$, $g = 4$, and $t(i) = 4$ for all nodes $i$.

Notice that although these performance results are for $W = 8$ $g = 4$, and $t(i) = 4$ they also apply to a network with $W = 2$, $g = 16$, and $t(i) = 1$ because for both cases there are 32 WDM channels per link and each node terminates at most 16 traffic streams in the clockwise or counter-clockwise directions of the ring. In fact, the performance results apply as long as $t(i) = W/2$ and $W \cdot g = 32$.

- Ring 2: For this ring $N = 16$, $W = 4$, and $g = 2$. In addition, it was assumed that for all nodes $i$, $t(i) = 1$. At nodes $\{0, 4, 8, 12\}$, all four wavelengths were terminated by ADMs. At nodes $\{2, 6, 10, 14\}$, three wavelengths $\{0, 1, 2\}$ were terminated by ADMs. For the rest of the nodes, only wavelength 0 was terminated by an ADM. Again this placement of ADMs is for the uniform traffic pattern where each pair of nodes will have on average one traffic stream between them.

  We ran simulations that stopped after one million conforming traffic streams were generated for different values of $a$. Figure 4.44 shows the average channel utilization for the ring for different values of $a$. The fraction of conforming traffic streams that were blocked is plotted in Figure 4.45. Note that the blocking is fairly low, at most a half a percent.

  We can make the conclusion from the simulations that the Incremental ring performs well for the conforming Poisson traffic. For Ring 1, there was no measured blocking, and for Ring 2, the blocking was less than 0.5%.

### 4.2.2 General Topologies

#### A Nonblocking General Topology Network

A nonblocking general topology network for semi-dynamic traffic satisfying Assumption B.2 is given in Appendix E [31]. We will refer to this as the *General Topology Optical Network.* The network is composed of optical and opaque nodes, such that each optical node only has opaque nodes as neighbors. An example network is shown in Figure 4.46. The number of wavelengths in the network is

$$W = L + \max_{i \in O} d(i) - 2$$

where $L$ is the traffic parameter of Assumption B.4, $O$ is the set of optical nodes, and $d(i)$ is the *degree* of node $i$ (i.e., the number of incident links). Notice that the network

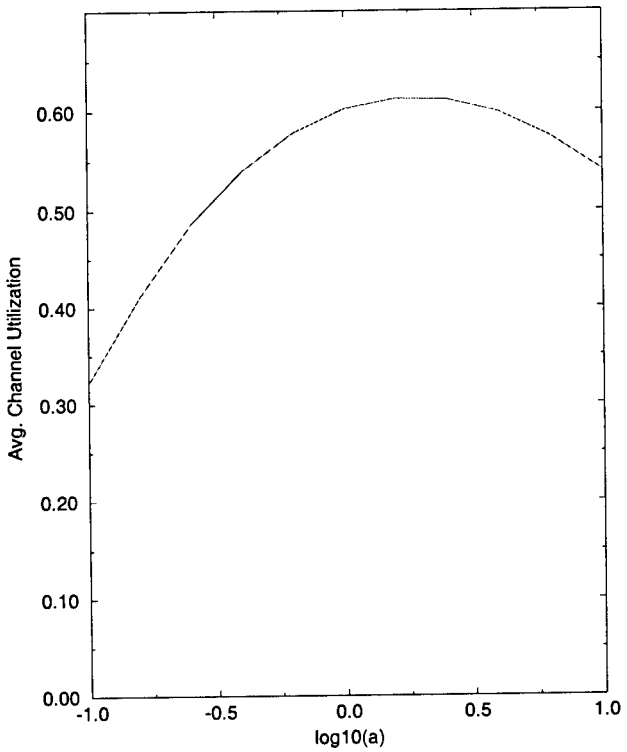



Figure 4.44: Average WDM channel utilization for a ring network with $N = 16$, $W = 4$, $g = 2$, and $t(i) = 1$ for all nodes $i$. The average channel utilization is a function of $log_{10}(a)$. For this plot, the value of $a$ ranges from 0.1 to 10.0.

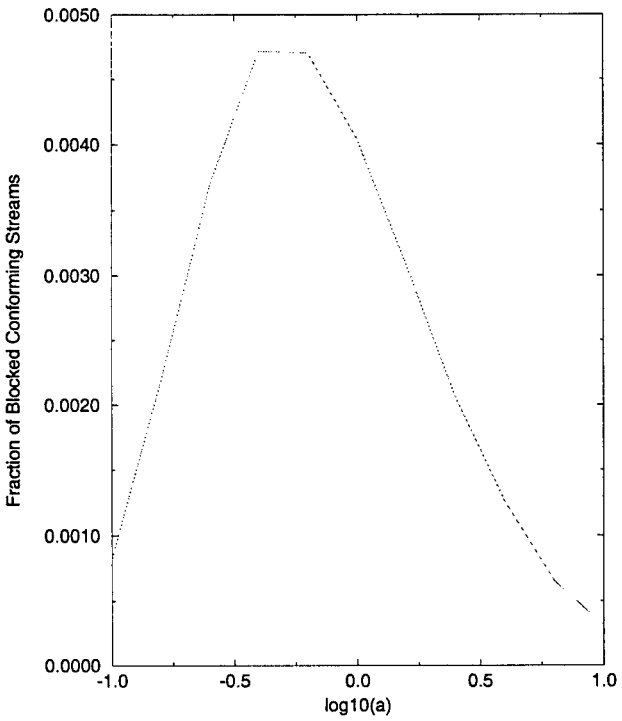



Figure 4.45: Fraction of conforming traffic streams that were blocked in a ring network with $N = 16$, $W = 4$, $g = 2$, and $t(i) = 1$ for all nodes $i$. The average channel utilization is a function of $log_{10}(a)$. For this plot, the value of $a$ ranges from 0.1 to 10.0.

requires $\max_{i \in O} d(i) - 2$ more wavelengths than a point-to-point network, i.e., a network that is only composed of opaque nodes. However, it requires few additional wavelengths for networks which have small degree. Fortunately, this is the case for backbone networks, e.g., NSFNET. Also notice that for a ring topology the General Topology Optical Network requires no additional wavelengths.

Figure 4.47 shows what kinds of cost savings are possible with the General Topology Optical Network over the point-to-point network. The figure shows simulation results for the NSFNET topology, Figure 4.46. Since each node in the topology has at most 4 incident links, the additional number of wavelengths is at most 2. The cost considered is the number of LTEs.

For the simulations, the traffic parameter $L = 32$ and the $\{t(i)\}$ values are chosen from an interval, randomly and uniformly distributed. For each set of $\{t(i)\}$, a minimum cost was computed by exhaustively examining all possible sets of optical nodes. The table in the figure has four columns. The first column gives the range of values of $\{t(i)\}$. Note that the average value of $\{t(i)\}$ is always 16, which is half of $L$. The second

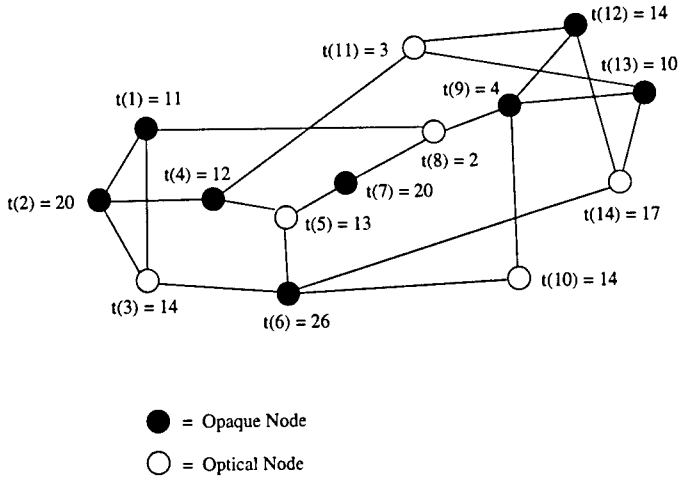



Figure 4.46: A General Topology Optical Network.

| Range of $t(i)$ | Min Cost | Reduced Cost | Percentage Reduction |
|---|---|---|---|
| [16..16] | 1168 | 176 | 13.1% |
| [14..18] | 1164 | 180 | 13.4% |
| [12..20] | 1152 | 192 | 14.3% |
| [10..22] | 1144 | 200 | 14.9% |
| [8..24] | 1140 | 204 | 15.2% |
| [6..26] | 1130 | 214 | 15.9% |
| [4..28] | 1120 | 224 | 16.7% |
| [2..30] | 1112 | 232 | 17.3% |

Figure 4.47: Table showing the average minimum cost for different ranges of values of $\{t(i)\}$.

column "Min Cost" gives the average minimum cost over 100 random samples. The third column "Reduced Cost" gives the average reduction in cost compared to a point-to-point network. Note that a point-to-point network will require $W = L$ wavelengths and 1344 LTEs to insure no blocking of offered traffic streams, under the link and node traffic constraints of Assumption B.4. The reduced cost is the minimum cost of the General Topology Optical Network minus 1344. The fourth column "Percentage Reduction" is the percentage of reduction, which is equal to $\frac{Reduced\ Cost}{Min\ Cost} \times 100\%$. Figure 4.47 shows that the more variability in $\{t(i)\}$ the greater the reduction in cost. Figure 4.46 shows an example solution when $\{t(i)\}$ is chosen within the range [2..30]. For this solution, the number of wavelengths required is $L + 1$, which is just one more than what is needed for a point-to-point network to insure no traffic blocking.

### A Design Algorithm for the General Topology Optical Network

We investigated the problem of designing a General Topology Optical Network to minimize cost. The problem is to determine which nodes should be optical or opaque. If the cost of a network is the number of LTEs then the problem is NP-Complete [31]. We show this in Appendix E which is a copy of [31].

In the appendix we also describe a simple design algorithm. It is a local search algorithm. It starts from a network that has only opaque nodes. Then it converts opaque nodes to optical ones and at all times the optical nodes will have only opaque nodes as neighbors. Each possible configuration $u$ of optical and opaque nodes leads to a network cost $f(u)$. In this case, the cost is the number of LTEs. The selection of which opaque node to convert to an optical one is random. However, there is a bias towards minimizing cost. This bias is dependent on a parameter $c$. The smaller $c$ is the larger the bias is to selecting a node that results in the minimum cost.

The appendix shows simulation results of how well the algorithm does in determining a good solution. The performance is shown to be sensitive to one of the parameters $c$ of the algorithm. Although it is not stated in the appendix, the design algorithm can be easily modified so that the cost function $f$ is now dependent on other components such as OXCs and DCSs.

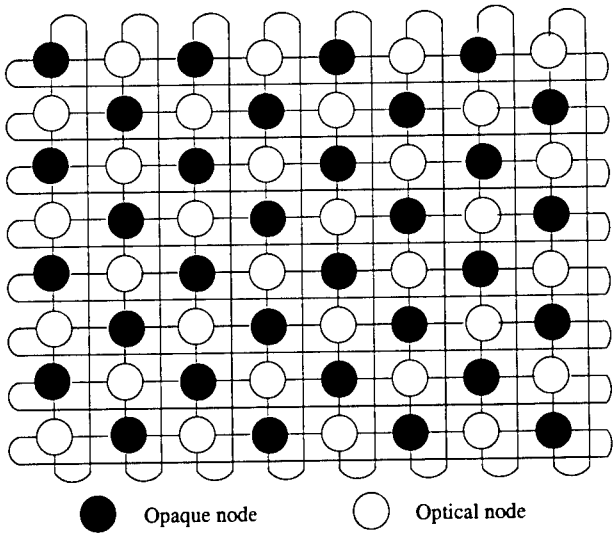



Figure 4.48: Optical network

### Simulations of Random Traffic

We chose the $8 \times 8$ mesh for our simulation topology, which is shown in Figure 4.48. This topology is sometimes referred to as a *torus*. It is simple and has good connectivity. Traffic inside the mesh has average length of at least four hops. Note that exactly half the nodes are optical nodes, which is the maximum possible for this topology.

Routing of traffic streams is assumed to be *XY routing*. This means that to route a traffic stream, start at one of its end nodes. Then route along a row to get to the column of its other end node. Then route along the column to the other end node. Routing along a row or column is always done using shortest hop paths. Note that when routing on a row or column, there may be more than one shortest hop path. Then ties are broken as follows. A tie is broken by the relative position of the two end nodes of traffic stream on the mesh topology in Figure 4.48. One of the end nodes is the *first* and the other is the *second*. To break a tie along a row, we check whether the first is on the right side of the second. Then the route goes to the right. Otherwise, it goes to the left. To break a tie along a column, we check whether the first node is below the second. Then we route upwards. Otherwise, we route downwards.

Besides its route along the fiber links, a traffic stream needs to be assigned to wavelengths. We will discuss how it is routed through optical nodes, which is sufficient to understand how it is routed through the network.

Now an optical node has its wavelengths partitioned into *local* and *transit* wavelengths.

At local wavelengths, all channels are terminated by LTEs. These channels are for *local traffic*, i.e., traffic that terminates at the node. At transit wavelengths, all channels are connected to a wavelength-selective OXC. The OXC is also connected to LTEs. Note that the OXC can be configured to terminate a channel at an LTE or it can connect channels to form optical pass-through. If the OXC terminates a channel with an LTE, we refer to this channel as a one-hop lightpath. If the OXC connects two channels, we refer to this as a two-hop lightpath. Transit wavelengths are primarily for transit traffic. Note that we also refer to the channels at local wavelengths as one-hop lightpaths too.

The routing of a traffic at the node is as follows:

- *Transit traffic:*
  - If available, send the traffic through a two-hop lightpath along the route. Use a partially filled one if possible.
  - Otherwise, send the traffic through one-hop lightpaths using the ones at transit wavelengths if possible. Also, of the transit wavelengths, use the partially filled ones first.
- *Local traffic:*
  - Use a one-hop lightpath at a local wavelength if possible.
  - Otherwise, use a one-hop lightpath at a transit wavelength.

The traffic for the simulation is random. However, it is with respect to the parameters given in Assumption B.4. First, the value $L = 16$. Since the topology has four incident links for each node, $W = 18$. Second, the values $\{t(i)\}$ are all the same. We denote this value by $t$.

The traffic stream is modified from an original Poisson process traffic. This original traffic has tributary traffic arriving as a Poisson process at rate 1. The two end nodes of the traffic are chosen randomly and uniformly over all possible pairs of nodes. If the traffic is set up in the network then its holding times are random and exponentially distributed with some mean. Of this original traffic, we discard those that violate the constraint that over any link there are at most $g \cdot L$ traffic streams. Thus, when a traffic stream for the original Poisson traffic arrives, it is XY routed on the mesh. Then if on any link there

already exists exactly $g \cdot L$ traffic streams then it is *discarded.* These discarded traffic will be referred to as *nonconforming.* The rest are considered *conforming* and they are attempted to be set up. If they cannot be set up then they are *blocked* by the network. Notice that if the network were a point-to-point network then there would be no blocking of conforming traffic.

Figure 4.49 shows the results of our simulations on the $8 \times 8$ mesh. Each data point is the result of 100,000 arriving traffic of original Poisson traffic. Notice that each column corresponds to different *mean hold times* (MHTs) of the traffic streams. Each column has an $LB$ value that equals the fraction of the original Poisson traffic that is nonconforming (discarded). This indicates the load in the network. If $LB$ is large then it means that the load must be high. The rest of each column are the $B$ values, which are the fraction of conforming traffic that is blocked. The $B$ values are for different values of $t$. As expected, the larger $t$ is the lower the blocking. Notice that in all cases of MHT, if $t = 9$ then the fraction of blocking of conforming traffic is less than 1%. If $t = 6$ then the blocking will increase, but below 2%. Therefore, with relatively small $t$ the General Topology Optical Network has low blocking, and therefore performs like a point-to-point network.

Now let us compare the LTE cost of this network with a point-to-point network. Suppose we want blocking of conforming traffic to be less than 1%. According to the figure, this means that $t = 9$. Then the average number of LTEs per node in the General Topology Optical Network is 63, which is slightly smaller than the average number of LTEs per node for the point-to-point network of 64. This is a modest improvement. However, if we want blocking of less than 2% then we only need to have $t = 6$. Then the average number of LTEs per node for the General Topology Optical Network is 60, as compared to 64 for the point-to-point network. Thus, the improvements are modest for this example. This should be expected since the degree of every node is 4 which is not very small. This keeps the General Topology Optical Network from exploiting a large fraction of optical pass-through.

| t | MHT = 75 | MHT = 150 | MHT = 200 | MHT = 300 |
|---|---|---|---|---|
| | $LB = 0.00000$ | $LB = 0.11201$ | $LB = 0.24046$ | $LB = 0.415476$ |
| | $B$ | $B$ | $B$ | $B$ |
| 3 | 0.04330 | 0.26200 | 0.26609 | 0.23335 |
| 4 | 0.00045 | 0.10805 | 0.13304 | 0.12890 |
| 5 | 0.00000 | 0.02536 | 0.03645 | 0.04970 |
| 6 | 0.00000 | 0.00992 | 0.01479 | 0.01668 |
| 7 | 0.00000 | 0.00800 | 0.01066 | 0.01167 |
| 8 | 0.00000 | 0.00724 | 0.00883 | 0.01155 |
| 9 | 0.00000 | 0.00687 | 0.00846 | 0.00855 |
| 10 | 0.00000 | 0.00660 | 0.00781 | 0.00775 |
| 16 | 0.00000 | 0.00000 | 0.00322 | 0.00720 |
| 18 | 0.00000 | 0.00000 | 0.00000 | 0.00000 |

Figure 4.49: Blocking rates ($MHT$ = Mean Holding Time)

# Chapter 5

# Conclusions

This project investigated WDM network that economized on switching. We did this at both the optical and tributary traffic layers. At the optical layer, our purpose was to investigate limited wavelength conversion. We investigated the design of CAPs that lead to good network performance. We also investigated efficient channel assignment algorithms for lightpaths. We were able to accomplish Tasks A.1-A.4 described in Section 3.1.

- **Task A.1.** We designed two channel assigment algorithms (DFS and Shortest Path) that assumed lightpaths come with pre-computed routes. It was shown that DFS and Shortest Path had comparable performances over a wide variety of CAPs. We also investigated four wavelength preferences for DFS and concluded that Fixed and Pack perform significantly better.

- **Task A.2.** We ran simulations on five types of CAPs to see if the conversion pattern can make an impact on performance while keeping w-degree fixed. We found that this was true, and that local CAPs (DISTRIBUTE, SHIFTED, and PARTITION) perform better than expanding ones (RANDOM-RANDOM, RANDOM-SAME, and SHUFFLE). We have the following reason of why this is true. Good channel assignments tend to keep long trails of free channels. Local CAPs may do a better job of this since they tend to connect channels at the same wavelength. This grouping of wavelengths locally may have the effect of creating thick "pipes" of connected free channels. It may be easier to find long trails of free channels in these pipes. On the other hand, expanding caps tend to spread connecting channels. Then the long trails of free channels may be easier to break.

We also showed by simulations that increasing w-degree to a modest amount can significantly reduce blocking. Though the reduction is not by orders of magnitude.

- **Task A.3.** We investigated two types of more sophisticated channel assignment algorithms. One type allows existing lightpaths to be rearranged to make way for new ones. This was RAS, Single-MTV, and Multiple-MTV. All assumed that new lightpaths come with pre-computed routes, and routes are never changed. Both RAS and Single-MTV rearranged only one existing lightpath to keep disruptions to a minimum. Our simulations show that they perform the same. This implies that MTV is a preferrable scheme since it has lower disruption of service than RAS.

  Multiple-MTV performs better than Single-MTV or RAS. For example, compared to no rearrangement, in one case, RAS, Single-MTV, and Multiple-MTV had average reduction of blocking by probabilities are significant, but not an overwhelming improvement. Thus, we can conclude rearrangement does improve performance, but the improvement may not warrant the more complicated schemes.

  The second type of channel assignment algorithms are for lightpath requests that do not come with their own pre-computed routes. Then the route along with the wavelengths along the route must be computed. We proposed three algorithms: Brute Force (modification of breadth first search), Shortest Distance, and Widest Shortest. Shortest Distance performed best. For limited wavelength conversion networks, Brute Force and Shortest Distance worked better than Widest Shortest because they find channel assignments with respect to CAPs at intermediate nodes, while Widest Shortest does not.

  In addition, some of our simulation results show that limited wavelength conversion (w-degree 4) can significantly improve the average time to first block over no conversion (one example showed improvement by 40%). However, for Shortest Distance, the improvements are not as great.

- **Task A.4.** We developed a simple design algorithm (DesignCAP) that finds optimal CAPs at particular w-degrees. The run time of the algorithm is prohibitive. However, it does find good CAPs, and can significantly improve existing ones. Our simulations also showed that a variation of the DISTRIBUTE CAP was locally

optimal in the sense that there is no perturbation of it that improves performance. This provides more evidence that that local CAPs lead to good performance.

At the tributary traffic layer we developed network architectures for ring and general topologies that guarantee no blocking of tributary traffic. These architectures require less LTE and switching equipment while providing the same or nearly the same bandwidth efficiency as point-to-point networks.

- **Task B.1.** We focused on ring topologies. We found network architectures Double-Hub, Hierarchical, and Incremental that are nonblocking. The Incremental ring is wide sense nonblocking for semi-dynamic traffic and uses the same amount of bandwidth as a point-to-point network. We developed a design algorithm for it that minimizes the ADM and DCS cost. From simulations, the design algorithm can significantly reduce ADM costs over point-to-point networks, and in some cases by an average of 44%.

  The Incremental ring is rearrangeably nonblocking for dynamic traffic, but not necessarily wide-sense nonblocking. We ran simultaions to see if there was significant blocking with random "typical" traffic. Our simulations show that there is low blocking of 0.5%.

- **Task B.2.** We tried to extend our results for rings to general topologies. We were successful, but in a limited way. We found the General Topology Optical Network that is nonblocking for semi-dynamic traffic, but it requires more wavelengths than a point-to-point network. The number of wavelengths is dependent on the maximum number of incident links on any node (i.e., maximum node degree). However, many backbone topologies have small node degree.

  By simulations, we showed that the General Topology Optical Network can lead to lower LTEs per node over a point-to-point network. For the NSFNET topology, our simulations showed a reduction in LTEs by as much as 17%.

  The network is rearrangeably nonblocking for dynamic traffic, but not necessarily wide-sense nonblocking. We ran simulations to determine the efficiency of the network for dynamic random traffic as compared to the point-to-point network.

We will finally mention that this project provided support and the primary research material for three Master's theses:

- Seemant Choudhary, *Routing of Dynamic Traffic on Wavelength Routing Networks*, M.S. Thesis Department of Electrical Engineering, University of Hawaii, May 2000.
- Ashutosh Gore, *Lightpath Rearrangement in WDM Networks*, M.S. Thesis, Department of Electrical Engineering, University of Hawaii, May 2000.
- Tao Lin, *Wavelength Division Multiplexed (WDM) Network with Minimal Transmitters and Receivers*, M.S. Thesis, Department of Electrical Engineering, University of Hawaii, December 1999.

# Bibliography


[1] D. Banerjee and B. Mukherjee, "A practical approach for routing and wavelength assignment in large wavelength-routed optical networks," *IEEE JSAC*, vol. 14, no. 5, June 1996, pp. 903-908.

[2] J. Bannister, L. Fratta, and M. Gerla, "Topological design of the wavelength-division optical networks," *Proc. IEEE INFOCOM '90*, 1990, pp. 1005-1013.

[3] J. Bannister, L. Fratta, and M. Gerla, "Topological design of wavelength-division optical network," *Proc. IEEE Infocom 90*, 1990, pp. 1005-1013.

[4] R. Barry and P. Humblet, "Models of blocking probability in all-optical networks with and without wavelength changes," *IEEE JSAC*, vol. 14, no. 5, June 1996, pp. 858-865.

[5] V. Benes, *Mathematical Theory of Connecting Networks.* Academic Press, 1965.

[6] C. Berge, *graphs and Hypergraphs.* North Holland 1976.

[7] D. Bienstock and O. Gunluk, "Computational experience with a difficult mixed-integer multicommodity flow problem," *Mathematical Programming,* vol. 68, pp. 213-237, 1995.

[8] T. Cormen, C. Leiserson, R. Rivest, *Introduction to Algorithms*, McGraw-Hill, 1990.

[9] A. M. Deguid, "Structural properties of switching networks," Brown University, Progress Report BTL-7, 1959.

[10] G. Ellinas, K. Bala, and G.K. Chang, "Scalability of a novel wavelength assignment algorithm for WDM shared protection rings," *Proc. Optical Fiber Communications (OFC) Conference, '98*, pp. 363-364, Feb. 1998.

[11] A. Elrefaie, "Multiwavelength survivable ring network architecture," *Proc. ICC*, pp. 1245-1251, 1993.

[12] A. Ganz and X. Wang, "Efficient algorithm for virtual topology design in multihop lightwave networks," *IEEE/ACM Trans. Networking*, vol. 2, pp. 217-225, June 1994.

[13] M. Garey and D. Johnson, *Computers and Intractibility: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Company, New York, 1979.

[14] O. Gerstel, P. Lin, and G. Sasaki, "A new angle on wavelength assignment in WDM rings: minimize system cost, not number of wavelengths," *Proc. IEEE Infocom 98.*

[15] O. Gerstel, R. Ramaswami, and G. Sasaki, "Cost effective traffic grooming in WDM rings," *Proc. IEEE Infocom 98*, April 1998.

[16] O. Gerstel, G. Sasaki, and R. Ramaswami, "Dynamic wavelength allocation in WDM ring networks with little or no wavelength conversion," *Proc. 1996 Allerton Conf. on Commun., Control, and Computing*, Oct. 1996, pp. 32-43.

[17] O. Gerstel, G. Sasaki, S. Kutten, and R. Ramaswami, "Worst-case analysis of dynamic wavelength allocation in optical networks," *IEEE/ACM Trans. Networking*, vol. 7, no. 6, pp. 833-843, December 1999.

[18] R. Guerin, A. Orda, and D. Williams, "QoS routing mechanisms and OSPF exensions," IETF Internet Draft, ¡draft-guerin-qos-routing-ospf-00.txt¿, Nov. 1996.

[19] J. Hui, *Switching and traffic theory for integrated broadband networks*, Norwell: Kluwer Academic, 1990.

[20] J.-F. Labourdette and A. Acampora, "Logically rearrangeable multihop lightwave networks," *IEEE Trans. Commun.,* vol. 39, pp. 1223-1230, Aug. 1991.

[21] K. Lee and V.O.K. Li, "A wavelength rerouting algorithm in wide-area all-optical networks," *IEEE/OSA J. Lightwave Technology*, vol. 14, no. 6, pp. 1218-1229, June 1996.

[22] A. Mohktar, M. Azizoglu, "Adaptive wavelength routing in all-optical networks," *IEEE/ACM Trans. Networking*, vol. 6, pp. 197-206, 1998.

[23] B. Mukherjee, S. Ramamurthy, D. Banerjee, and A. Mukherjee, "Some principles for designing a wide-area optical network," *Proc. IEEE INFOCOM '94*, 1994.

[24] E. Modiano and A. Chiu, "Traffic grooming algorithms for minimizing electronic multiplexing costs in unidirectional SONET/WDM ring networks," *Proc. CISS '98, to appear.*

[25] C. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*, Englewood Cliffs: Prentice-Hall, 1982.

[26] R. Ramaswami and G. Sasaki, "Multiwavelength optical networks with limited wavelength conversion," *IEEE/ACM Trans. Networking*, vol. 6, no. 6, pp. 744-754, December 1998.

[27] R. Ramaswami and K. Sivarajan, "Design of logical topologies for wavelength-routed optical networks," *IEEE JSAC*, vol. 14, no. 5, pp. 840-851, June 1996.

[28] P. Roorda, C.-Y. Lu, and T. Boutilier, "Benefits of all-optical routing in transport networks," *OFC '95 Conf. Edition Technical Digest*, vol. 8, pp. 164-165, 1995.

[29] G. Sasaki, O. Gerstel, and R. Ramaswami, "A WDM ring network for incremental traffic," *Proc.* $36^{th}$ *Annual Allerton Conference*, Monticello, IL, pp. 662-672, Sept. 1998.

[30] J. Simmons, E. Golstein, and A. Saleh, "On the value of wavelength-add/drop in WDM rings with uniform traffic," *Proc. OFC '98*, pp. 361-362, Feb. 1998.

[31] G. Sasaki and T. Lin, "A minimal cost WDM network for incremental traffic," *Proc. SPIE Conference on All-Optical Networking 1999: Architecture, Control, and Management Issues*, Boston, MA, Sept. 1999.

[32] R. Sedgewick, *Algorithms in C++*, Addisen-Wesley Publishing Co., 1992.

[33] D. Slepian, "Two theorems on a particular crossbar switching," unpublished manuscript, 1952.

[34] Y. Yates, J. Lacey, D. Everitt, and M. Summerfield, "Limited -range wavelength translation in all-optical networks," *Proc. IEEE Infocom '96*, pp. 954-961, 1996.

[35] X. Zhang and C. Qiao, "On scheduling all-to-all personalized connections and cost-effective designs in WDM rings," *Proc. of the SPIE Conf. on All-Optical Communication Systems: Architectures, Protocols and Network Issues II* vol. 2919, Nov. 1996.

# Appendix A

# Channel Assignment Simulations for NSFNET

In this appendix, we present the results of simulations for the NSFNET topology, w-degree 2, DFS, and the six CAPs. The results are are shown in Figures A.1, A.2, A.3, A.4, A.5, and A.6. They will be used to compare the performances of the wavelength preferences FIXED, PACK, SPREAD, and RANDOM.

Figure A.1 shows the blocking probability curves for PARTITION. Note that the figure also includes the curve for the case of no conversion (with PACK). Notice that FIXED and PACK have the lowest blocking probabiities, while SPREAD and RANDOM have much higher blocking probabilities. SPREAD and RANDOM have higher blocking than even the no conversion case. Figures A.2 and A.3 show the blocking probabilities for two other local CAPs, SHIFTED and PARTITION, respectively. The results do not change.

Figures A.4, A.5, and A.6, have the blocking probabilities for the expanding CAPS RANDOM-SAME, RANDOM-RANDOM, and SHUFFLE, respectively. PACK seems to work reasonably well for RANDOM-SAME and SHUFFLE, and not too poorly for RANDOM-RANDOM. The blocking probabilities are high for all expanding CAPs and wavelength preferences. To be more specific, the blocking probabilities are often worse than if the network had no conversion. Thus, the comparison of the wavelength preferences may be irrelevant for these expanding CAPs, and we should restrict our conclusions to the case of local ones.

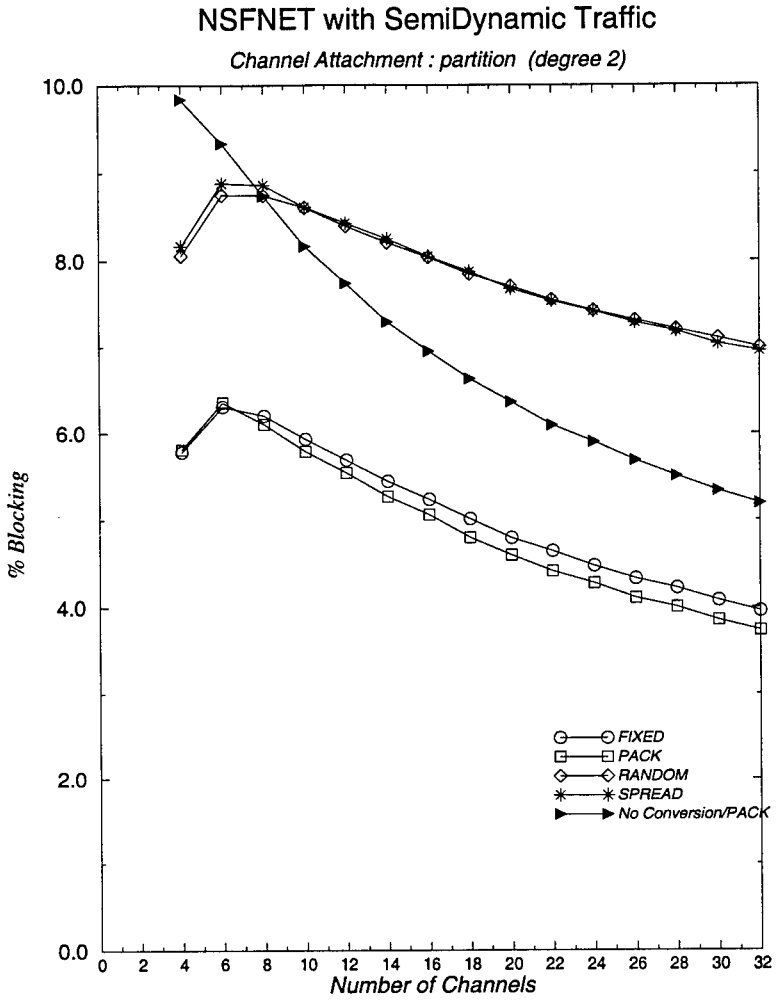



Figure A.1: NSFNET, w-degree 2, PARTITION.

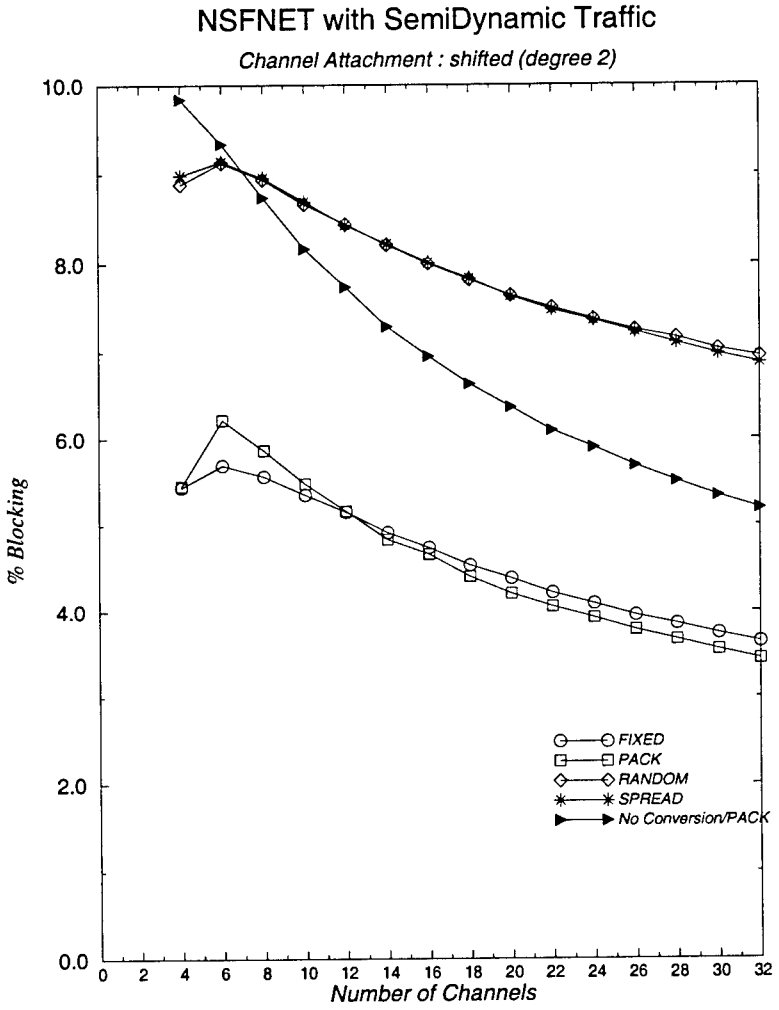



Figure A.2: NSFNET, w-degree 2, SHIFTED.

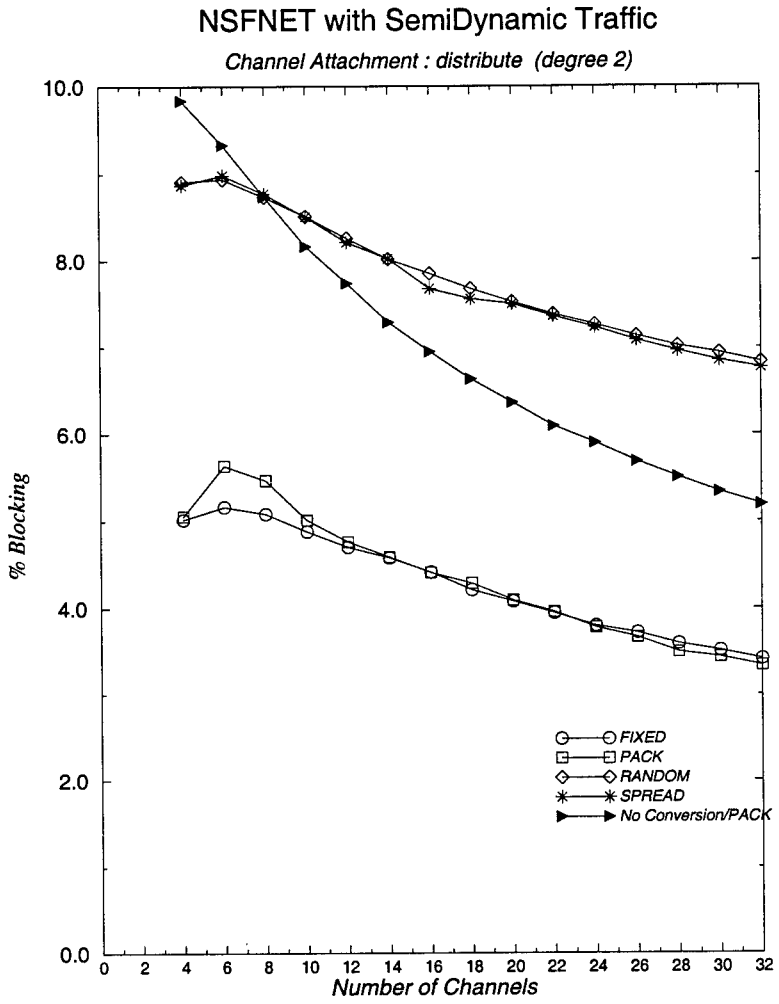



Figure A.3: NSFNET, w-degree 2, DISTRIBUTE.

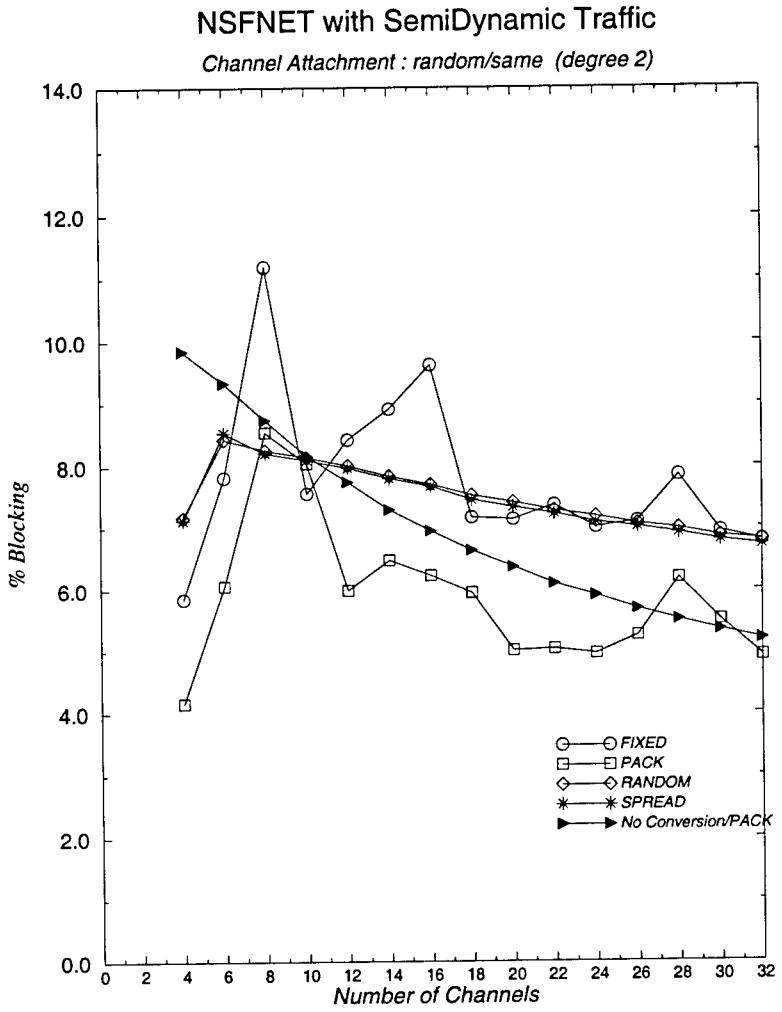



Figure A.4: NSFNET, w-degree 2, and RANDOM-SAME.

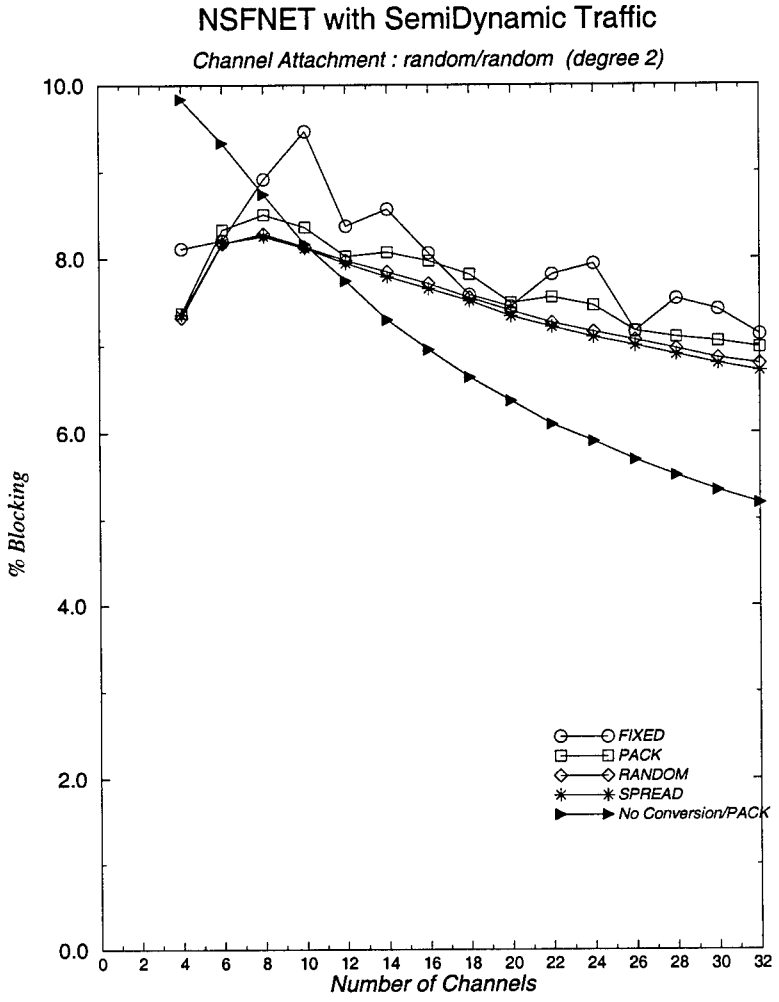



Figure A.5: NSFNET, w-degree 2, and RANDOM-RANDOM.

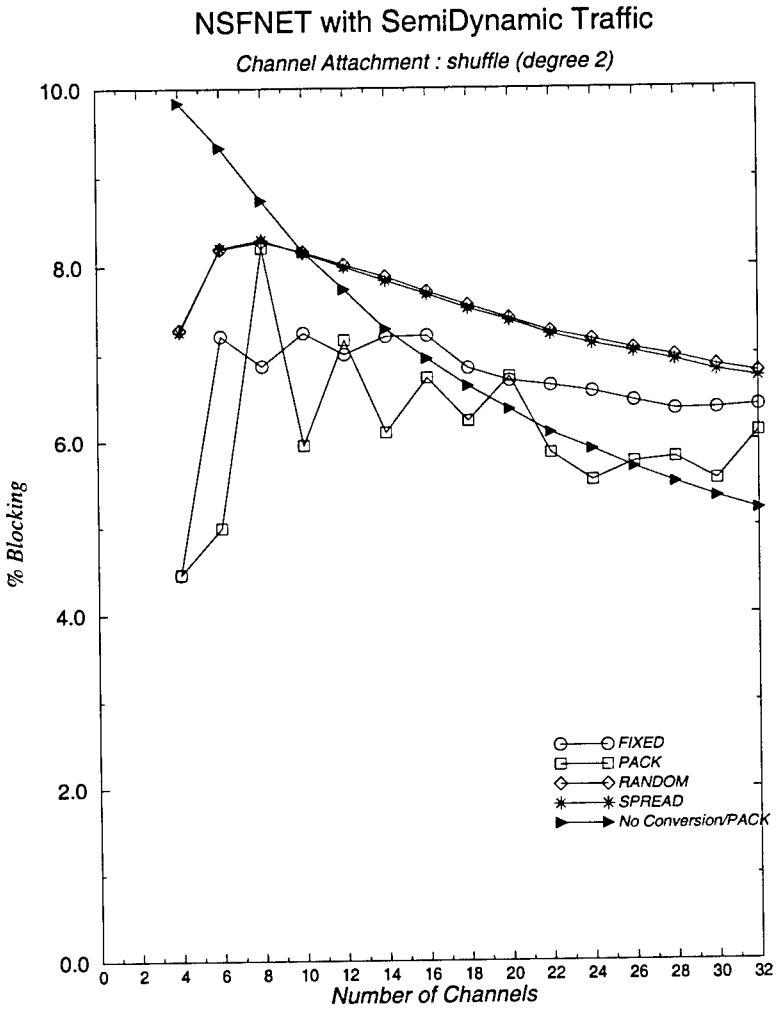



Figure A.6: NSFNET, w-degree 2, and SHUFFLE.

# Appendix B

# Channel Attachment Comparisons

In this appendix, we will present results that will compare the performance of different CAPs. The simulations are for the ring and NSFNET for w-degree 4, and a number of other topologies for w-degree 2. In all cases, the depth first search channel assignment was used with either FIXED or PACK. The results verify that local CAPs generally work better than expanding CAPs, and that DISTRIBUTE seems is best or nearly the best.

Before we discuss the results, we should note that performance for SHIFTED was not optimized. In other words, the determination of which nodes should have the PARTITION CAP, and which should have the S-PARTITION CAP was done arbitrarily rather than to optimize performance. In particular, nodes that were numbered with an odd value had the S-PARTITION, while nodes that were even valued had the PARTITION CAP. However, we do not believe any optimization will lead to large performance gains as we had discussed in Section 4.1.2.

Figures B.1 and B.2 show results for the ring and NSFNET, respectively, for w-degree 4, and for FIXED. The local CAPs work better than expanding ones, and DISTRIBUTE works best in both cases.

Next we consider a number of sparse topologies. These are two star networks in Figure B.3, and two trees shown in Figures B.4 and B.5. These topologies are labeled Star 1, Star 2, Tree 1, and Tree 2.

Figure B.6 shows the blocking probabilities for the sparse topologies including the 16 node ring for the case of 16 wavelengths and w-degree 2. The channel assignment used was depth first search and PACK. The X-axis corresponds to the different topologies, where the X-axis values of 1, 2, 3, 4, and 5, correspond to the ring, Star 1, Star 2, Tree 1, and Tree 2, respectively. The blocking probabilities of the local CAPs are better than

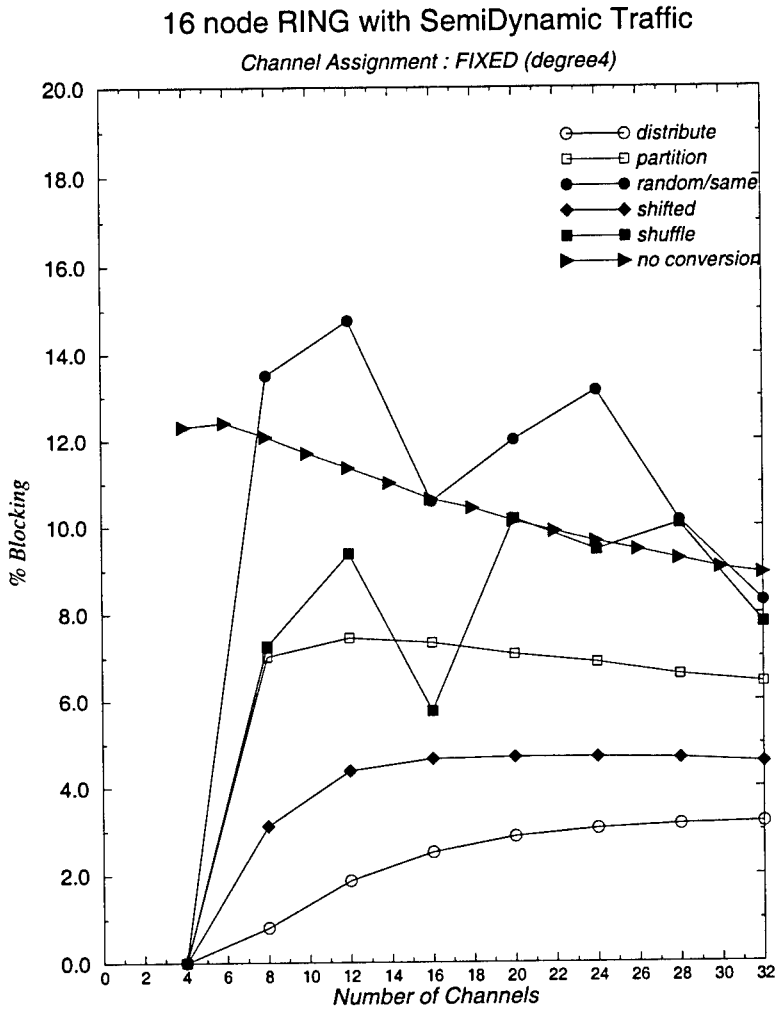



Figure B.1: Blocking probabilities for the 16 node ring, w-degree 4, and FIXED.

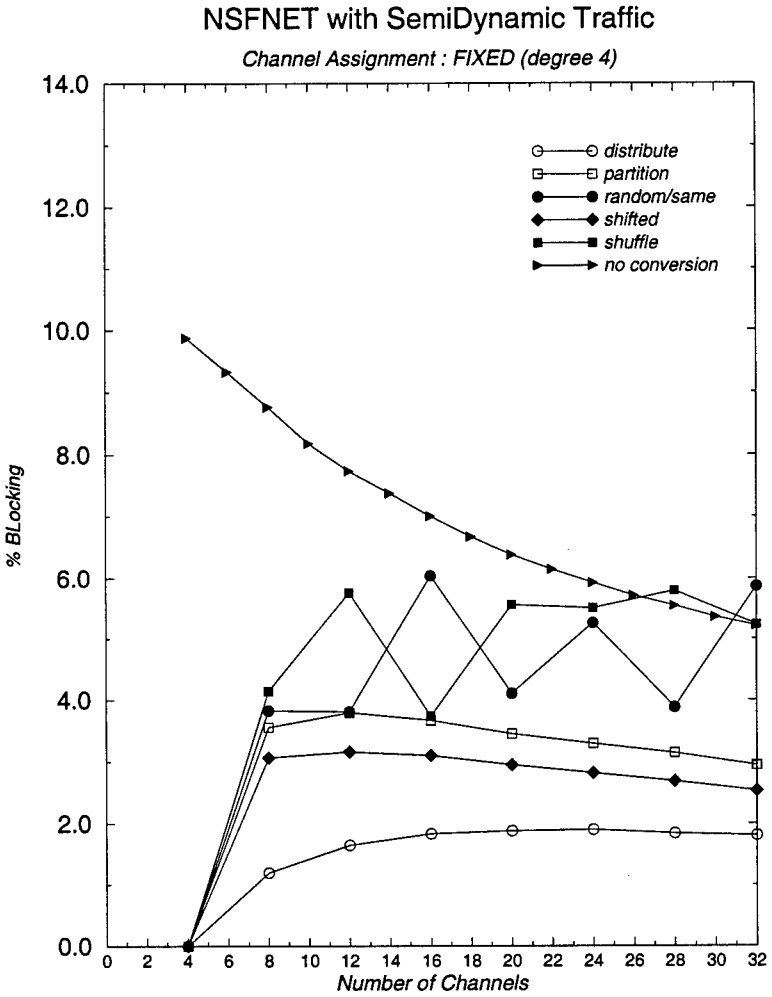



Figure B.2: Blocking probabilities for the 16 node ring, w-degree 4, and PACK.

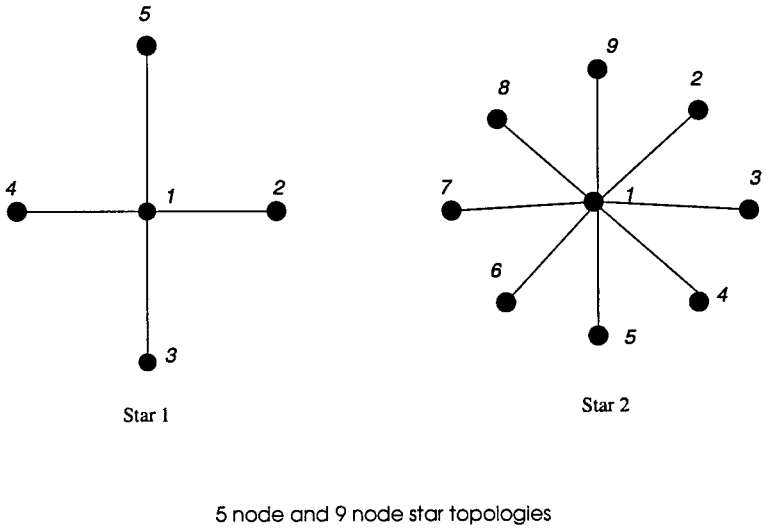
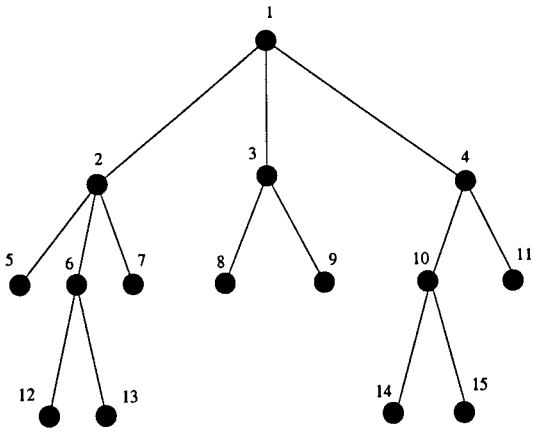
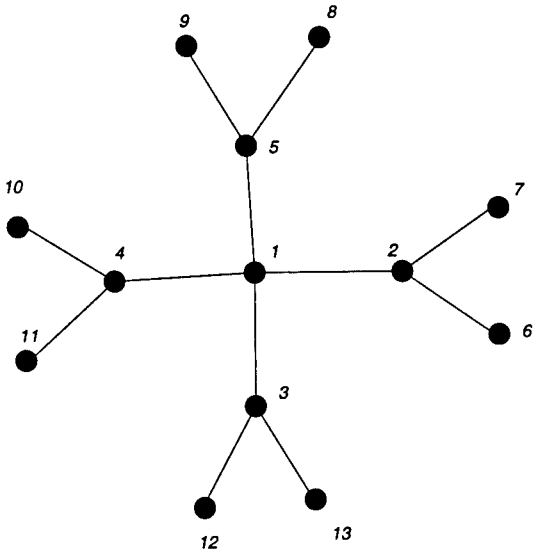



Figure B.3: Stars 1 and 2.




Tree Topology 1

Figure B.4: Tree 1.




Tree Topology 2

Figure B.5: Tree 2.

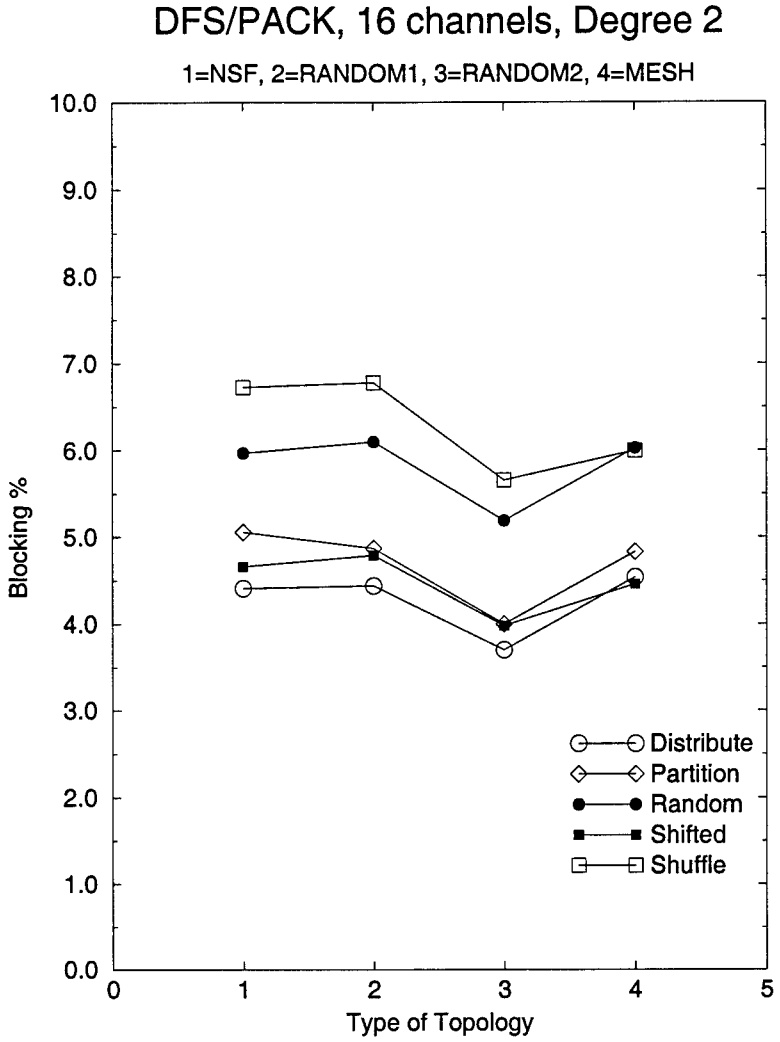



Figure B.6: The Mesh topologies for 16 wavelengths.

the expanding ones. In addition, the blocking probabilities of the local CAPs are close. Note that the CAP that performs the best depends on the topology. For example, for the ring, DISTRIBUTE works better than PARTITION, but for Star 1, it is the other way around.

Finally, we consider a collection mesh topologies. These are two random topologies in Figures B.7 and B.8, and the mesh connected cycles in Figure B.9. These topologies are labeled Random 1, Random 2, and Mesh.

Figure B.10 shows the blocking probabilities for the sparse topologies including NSFNET for the case of 16 wavelengths and w-degree 2. The channel assignment used was depth first search and PACK. The X-axis corresponds to the different topologies, where the X-axis values of 1, 2, 3, and 4, correspond to NSFNET, Random 1, Random 2, and Mesh

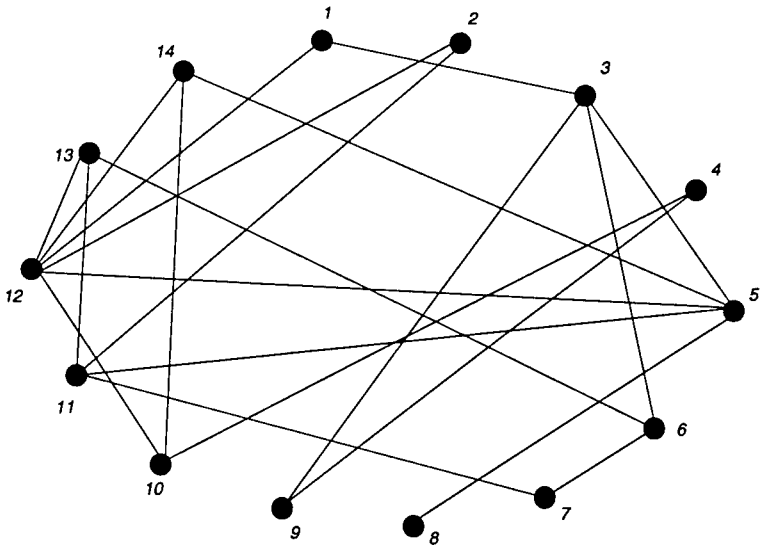



Figure B.7: Random 1.

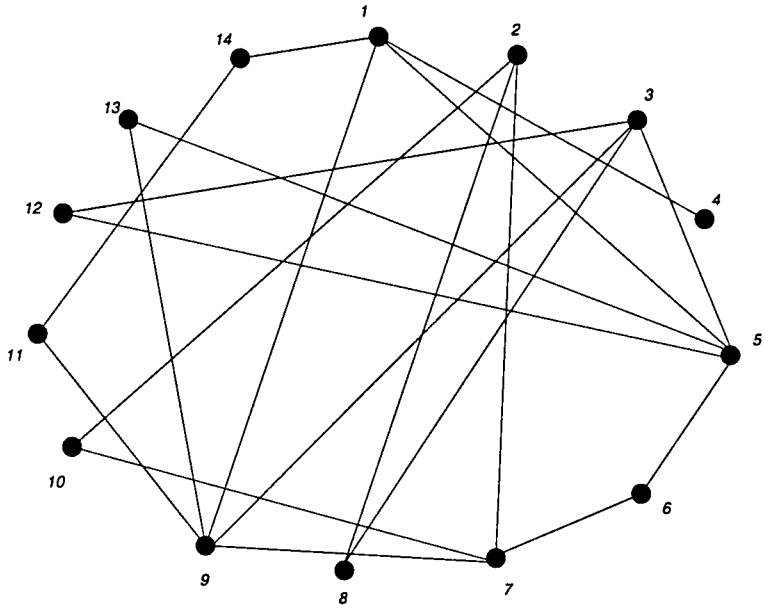



Figure B.8: Random 2.

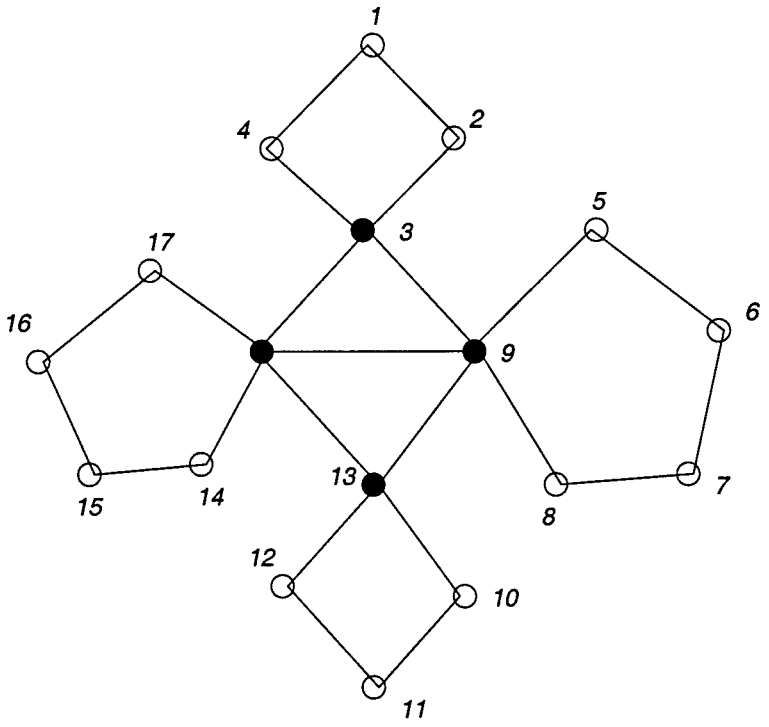



Figure B.9: Mesh connected rings.

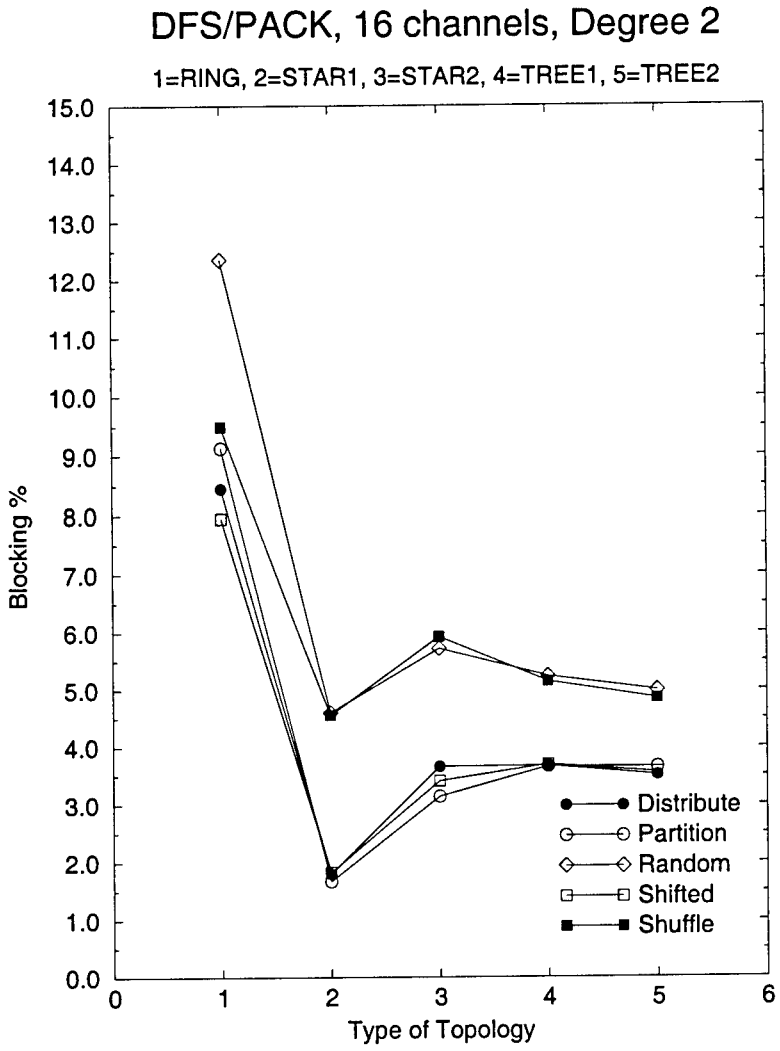



Figure B.10: The mesh topologies for 16 wavelengths.

Connected Rings, respectively. Again, the blocking probabilities of the local CAPs are better than the expanding ones, and DISTRIBUTE works best all the time.

# Appendix C

# Cost Effective Traffic Grooming in WDM Rings

Ornan Gerstel, Rajiv Ramaswami, Galen Sasaki[1]

**An earlier version of this paper appeared in the Proceedings of the IEEE Infocom 98.**

**Abstract.** We provide network designs for *optical add-drop wavelength division multiplexed* (OADM) rings that minimize overall network cost, rather than just the number of wavelengths needed. The network cost includes the cost of the transceivers required at the nodes as well as the number of wavelengths. The transceiver cost includes the cost of terminating equipment as well as higher-layer electronic processing equipment, which in practice can dominate over the cost of the number of wavelengths in the network. The networks support dynamic (i.e., time-varying) traffic streams that are at lower rates (e.g., OC-3, 155 Mb/s) than the lightpath capacities (e.g., OC-48, 2.5 Gb/s). A simple OADM ring is the *point-to-point* ring, where traffic is transported on WDM links optically, but switched through nodes electronically. Although the network is efficient in using link bandwidth, it has high electronic and opto-electronic processing costs. Two OADM ring networks are given that have similar performance but are less expensive. Two other OADM ring networks are considered that are *nonblocking*, where one has a *wide sense nonblocking* property and the other has a *rearrangeably nonblocking* property. All the networks are compared using the cost criteria of number of wavelengths and number of transceivers.

[1]This work was done in part while being with the Tellabs Operations. It was sponsored in part by grant MDA-972-95-C-0001 by DARPA, and was sponsored in part by grant F30602-97-1-0342 by DARPA and Rome Laboratory, Air Force Materiel Command, USAF. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes nothwithstanding any copyright annotation thereon. The views and conclusions contained therein are those of the authors and should not be expressed or implied, of the Defense Advanced Research Projects Agency (DARPA), Rome Laboratory, or the U.S. Government.

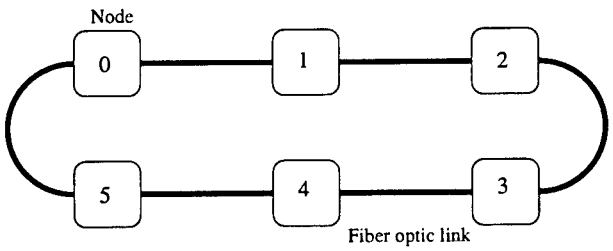



Figure C.1: An optical WDM ring.

**Keywords:** Optical networks, wavelength division multiplexing (WDM), lightwave networks, ring networks, network design.

# C.1 Introduction

An *optical add-drop wavelength-division-multiplexed* (WDM) ring network (OADM ring in short), shown in Figure C.1, consists of $N$ nodes labeled $0, 1, ..., N-1$ in the clockwise direction, interconnected by fiber links. Each link carries high-rate traffic on optical signals at many wavelengths. The network has a fixed set of wavelengths for all links which we denote by $\{w_0, w_1, ..., w_{W-1}\}$, where $W$ denotes the number of wavelengths. OADM ring networks are being developed as part of test-beds and commercial products, and are expected to be an integral part of telecommunication backbone networks. Although mesh topology WDM networks will be of greater importance in the future, at least in the near term, ring topologies are viable because SONET/SDH self-healing architectures are ring oriented.

OADM rings support *lightpaths*, which are all-optical communication connections that span one or more links. We will consider networks where each lightpath is full duplex, and its signals in the forward and reverse direction use the same wavelength and route. Since each lightpath is full duplex, it is terminated by a pair of *transceivers*. Here, a transceiver is generic for such systems as *line terminating equipment* (LTE) and *add/drop multiplexers* (ADM) (or more accurately, half an ADM). All lightpaths have the same transmission capacity, e.g., OC-48 (2.5Gb/s) rates.

A node in a OADM ring is shown in Figure C.2. Note that some of the lightpaths pass through the node in optical form. They carry traffic not intended for the node. The remaining lighpaths are terminated at the node by transceivers, and their traffic is converted to electronic form, and processed electronically. The electronic processing (and switching) includes systems such as SONET/SDH ADMs, IP routers, and *digital crossconnect systems* (DCSs), that cross connect traffic streams. To simplify the presen-

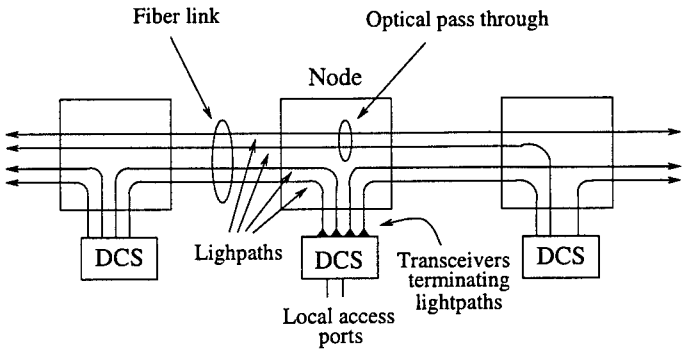



Figure C.2: An optical node.

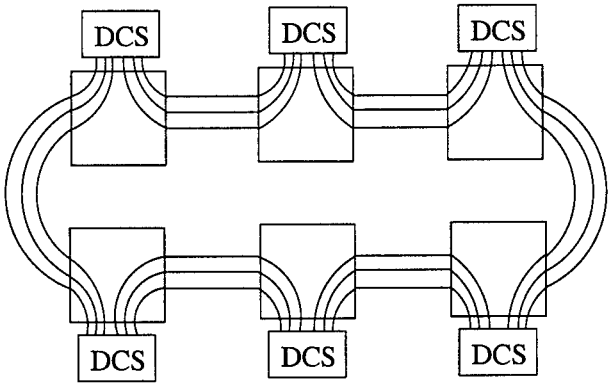



Figure C.3: A point-to-point OADM ring with three wavelengths.

tation, we shall assume DCS systems in the sequel, but the very same discussion holds for the other type of electrical nodes. In the figure, the DCS is shown representing all the electronic processing, and the transceivers are located at the interface of the DCS and lightpaths. Now some of the received traffic may be intended for the node, in which case it is switched to a local entity through local access ports. The rest of the traffic is forwarded on other lightpaths via the transceivers. In our model, the cost of transceivers is a dominant cost.

A special case of an OADM ring network is the *point-to-point* WDM ring network (PPWDM ring in short) shown in Figure C.3. Here, each link in the network has one-hop lightpaths on each of its wavelengths. The network is called a point-to-point ring because each lightpath implements a point-to-point connection between neighboring nodes. For the network, each node has a single DCS that cross connects traffic from all the lightpaths. The DCS is *wide sense nonblocking*, which means that a traffic stream may be routed through it without disturbing existing traffic streams. Note that this network does not have a true optical node because lightpaths do not pass through nodes, i.e., traffic at each node is processed electronically.

The PPWDM ring has the advantage of being able to efficiently use the link bandwidth for time-varying traffic. The network can route a traffic stream through it without disturbing other traffic streams as long as there is enough spare capacity along each link of the route. Hence, due to its capability to switch traffic streams between spare capacity on different wavelengths, it will be wavelength efficient. Its disadvantage is that its nodes do not have optical pass through, resulting in maximum transceiver cost. For instance, in a typical carrier network, each link may have 16 wavelengths, each carrying OC-48 data. Suppose an OADM ring node needs to terminate only one lightpath worth of traffic. In this case, the node would ideally pass through the remaining 15 lightpaths in optical form without "processing" them. On the other hand, a PPWDM ring would require the traffic from all 16 wavelengths to be received, possibly switched through an electronic DCS, and retransmitted.

In practice however, the situation is somewhat more complicated. Each lightpath typically carries many multiplexed lower-speed traffic streams (e.g., OC-3 streams, which are at 155 Mb/s). An OADM ring node cannot extract an individual lower-speed stream from a wavelength without first receiving the entire wavelength. Thus, in the example above, if we had to extract an individual OC-3 stream from each of the 16 wavelengths at a node, and all the remaining traffic were not intended for that node, all 16 wavelengths must be received. Note that the problem of designing networks that efficiently *grooms traffic* (i.e., multiplex/demultiplex lower-speed traffic streams onto and off-of higher capacity lighpaths) is nontrivial, and its solution can have a great impact on network cost.

### C.1.1 Design assumptions and approach

In this paper we will address the problem of designing OADM rings for cost effective traffic grooming. Our approach will be to propose and analyze a collection of OADM ring networks under the following assumptions and criteria:

1. **Network costs will be dealt with explicitly.** The costs of interest are

   - *Number of Wavelengths $W$.*
   - *Transceiver Cost $Q$:* The cost $Q$ is defined to be the average number of transceivers per node in the network. As it turns out, transceiver cost may reflect actual costs better than the number of wavelengths. Note that $Q$ is equal to twice the average number of lightpaths per node since two transceivers terminate each lightpath.

- *Maximum Number of Hops H:* The cost $H$ is defined to be the maximum number of hops of a lightpath. It is desirable to minimize $H$ since it leads to simpler physical layer designs.

While most of the previous work on WDM networks dealt with minimizing the number of wavelengths, our work, which first appeared in [15], is the first to consider transceiver costs. In addition, our cost analyses give formulas that quantitatively relate network resources with traffic parameters.

2. **Lightpaths are fixed, although their placement may be optimized at start up.** This is a reasonable assumption for practical WDM networks at least in the near term because (i) the traffic in a lightpath is an aggregation of many traffic streams, making it less likely to fluctuate significantly; (ii) automatic network switching for lightpaths is not yet cost effective; and (iii) rerouting lightpaths may cause disruption of service.

3. **The networks are circuit-switched, and support lower-speed, full-duplex end-to-end connections, all at the same rate.** For example, the lightpaths may be at the OC-48 rate and support only OC-3 circuit-switched connections. We will refer to these connections as *traffic streams.* We will let $c$ denote the number of traffic streams that can be supported in a lightpath, i.e., $c$ traffic streams = 1 lightpath. For example, for OC-48 lighpaths and OC-3 traffic streams, $c = 16$.

4. **Each node has a wide-sense nonblocking DCS that is large enough to crossconnect all traffic between its transceivers and local ports.** This assumption is realistic for many practical situations, and will simplify our subsequent discussion. Notice that the cost of the DCS is not considered in this paper. This is reasonable assuming that the interface-ports rather than switch-fabric dominate DCS costs because then total DCS cost is proportional with total transceiver cost.

The overall network design problem comprises of two phases: first the lower-speed traffic must be aggregated on to lightpaths, so as to minimize transceiver costs as well as wavelength costs. This is the focus of our paper. The second phase may incorporate constraints in organizing the lightpaths. For instance, an OADM network may be called upon to realize multiple SONET rings. This phase of network design is treated in a follow-on paper that also includes transceiver (ADM) costs [14]. Here, an OADM network must realize multiple SONET rings (one ring per wavelength). However, the lightpaths are already assumed to be given and the focus is on arranging them in rings. Besides [14], the

only other studies that consider transceiver costs are [24, 30], which focus on ring networks without DCSs and for specific *static* (i.e., fixed over time) traffic, e.g., uniform static traffic. Typically, researchers have concentrated on numbers of wavelengths, congestion, delay, or probability of blocking. We should mention that there is previous work on WDM network design for lower-speed traffic streams [2, 7, 12, 27, 28], assuming static traffic. There are also a number of papers on WDM networks with *dynamic* (i.e., time-varying) traffic (e.g., [4, 1, 17, 20, 26]), but where lightpaths are switched and lower speed traffic streams are not considered. The study of (non-statistical) dynamic traffic and fixed lightpaths for OADM networks seems to be unique to this paper.

### C.1.2 Traffic models

When considering a network architecture, the traffic time dependent behavior, distribution, and routing are of paramount importance. We consider three traffic types insofar as their time dependency is concerned: *static*, *dynamic*, and *incremental*. *Static* traffic means that lower-speed traffic streams are set up all at once, at some initial time, and fixed thereafter. *Dynamic* traffic means that traffic streams are set up and terminated at arbitrary times. *Incremental* traffic is dynamic traffic but where traffic streams never terminate. This models the situation when traffic streams are expected to have a long holding times, as is usually the case with provisioning of high-speed connections today.

The traffic distribution will be represented by a traffic matrix $T = [T(i,j)]$, where $c \cdot T(i,j)$ equals the number of traffic streams between nodes nodes $i$ and $j$. Thus, $T(i,j)$ is the number of "lighpaths of traffic" between nodes $i$ and $j$. Note that $T(i,j)$ can be fractional. For example, if 24 OC-3 connections (1 OC-48 = 16 OC-3s) are to be supported between $i$ and $j$ then $T(i,j) = 1.5$. If the traffic is static then $T$ is fixed for all time, while if the traffic is dynamic then $T$ is time-varying. Note that placement of transceivers is dependent on the traffic pattern. For example, for each node $i$, $\sum_{j=0}^{N-1} T(i,j)$ is a lower bound on the number of transceivers it requires.

The routing of traffic affects the traffic loads on links, which in turn affect bandwidth requirements. We consider traffic that either requires routing or are *pre-routed*, i.e., they come with their own pre-computed routes. In addition, pre-routed traffic are assumed to have *simple* routes, which means that they visit a node at most once. In this sense, they are routed efficiently in the network. Note that the pre-routed traffic model holds for many practical scenarios, such as when traffic is routed according to shortest paths or traffic-loads. It allows us to define a maximum "traffic load" over links, which is a lower bound on the number of wavelengths to accommodate the traffic.

We consider three different traffic assumptions (i.e., scenarios), given below. The first assumes dynamic traffic that only has restrictions on the amount of traffic that terminates at the nodes. The next assumption has pre-routed traffic and a maximum traffic load parameter. The parameter is a measure of the required bandwidth (wavelengths) on the links. This model may be more appropriate when wavelengths are limited because then the load parameter value can be chosen appropriately. The final assumption is a uniform traffic assumption used in the literature as a benchmark to compare different architectures.

**Traffic Assumption A.** Traffic is dynamic, i.e., $T$ is time-varying. The traffic has integer parameters $(t_A(i) : i = 0, 1, ..., N-1)$. At any time, each node $i$ can terminate at most $c \cdot t_A(i)$ traffic streams. Thus, at any time, for each node $i$, $t_A(i) \geq \sum_{j=0}^{N-1} T(i,j)$ and $t_A(i) \geq \sum_{j=0}^{N-1} T(j,i)$. □

**Traffic Assumption B.** The traffic is dynamic, with traffic streams being pre-routed and having simple routes. The traffic has integer parameters $L$ and $(t_B(i) : i = 0, 1, ..., N-1)$.

At any time, the number of traffic streams over any link is at most $c \cdot L$, assuming no blocking. In addition, each node $i$ may terminate at most $c \cdot t_B(i)$ traffic streams from the clockwise or counter-clockwise direction along the ring. Thus, node $i$ can terminate up to $2c \cdot t_B(i)$ traffic streams, but then half must come from the clockwise direction and the other half must come from the counter-clockwise direction. (Note that $2t_B(i)$ is a lower bound on the number of transceivers at node $i$ to insure no blocking.) □

**Traffic Assumption C.** This is the *static uniform traffic.* It has an integer parameter $g$, and has exactly $g$ traffic streams between every pair of nodes. Thus, $T(i,j) = g/c$ if $i \neq j$, and $T(i,j) = 0$ if $i = j$. This traffic is commonly used to compare networks in the theoretical literature because (i) it requires good network connectivity since all nodes are connected to one another, and (ii) its uniformity simplifies analysis.

### C.1.3 Proposed network architectures

In this paper, we will consider six OADM ring networks. To define these networks we need to specify the following: placement of lightpaths (and the corresponding transceivers) and a routing algorithm for the traffic streams onto lightpaths. Note that the placement of a lightpath requires finding a route and wavelength for it.

The six networks assume different constraints on the traffic. The following are brief descriptions of each OADM ring and their corresponding traffic constraints. In Section C.2 we will provide a more detailed description of the networks and their costs.

The following network assumes static traffic.

**Fully-Optical Ring:** For this network, between each pair of nodes $i$ and $j$ there are $\lceil T(i,j) \rceil$ lightpaths between them. Traffic streams between the nodes are carried directly by these connecting lightpaths. We consider this network because it has no electronic traffic grooming (which is why it is called "fully-optical"). It is therefore the opposite of the PPWDM ring which has maximal traffic grooming capability. Note that it becomes bandwidth efficient if the traffic is high enough to fill the lightpaths.

The next two networks are for Traffic Assumption $A$. Under the assumption, they are *nonblocking*, which means that they will not block any arriving traffic stream. Note that $t_A(i)$ is a lower bound on the number of transceivers at node $i$ to insure no blocking.

**Single-Hub:** This network has a unique node designated as a *hub*, which has lightpaths directly connecting it to all other nodes. It is *wide sense* nonblocking, i.e., traffic streams may be added without disturbing existing ones.

**Double-Hub:** This network has two hubs, which have lightpaths connecting them to all other nodes. This network is *rearrangeably* nonblocking, which means that it may have to rearrange existing traffic streams to make way for new ones. Note that rearranging existing traffic streams is undesirable in practical networks. However, the double-hub network is reasonably efficient in $W$ and $Q$, so it could be used for static traffic.

The single- and double-hub networks are nonblocking for dynamic traffic with the only constraint that each node $i$ can terminate at most $c \cdot t_A(i)$ traffic streams. They result in large $W$ to accommodate worst case traffic distributions that lead to high traffic loads on links. The next three ring networks are for Traffic Assumption B. They require $W \geq L$ to insure no traffic blocking. Notice that the requirement is necessary for any network to be nonblocking. However, the inequality by itself is insufficient because traffic cannot use spare bandwidth at different wavelengths if they cannot be switched at intermediate nodes.

**PPWDM Ring:** This is the PPWDM ring network described earlier. For Traffic Assumption B and $W = L$, it is wide-sense nonblocking.

**Hierarchical Ring:** This is a simple network composed of two PPWDM subrings, and is wide-sense nonblocking for Traffic Assumption B. The network uses more wavelengths than a PPWDM ring, but it often uses less transceivers.

**Incremental Ring:** This a ring network that is organized (recursively) from sections of the ring. For Traffic Assumption B, the network is wide sense nonblocking for incremental traffic. It requires the same $L$ wavelengths as the PPWDM ring, but a smaller number of transceivers. Since it is wide sense nonblocking for incremental traffic, it is rearrangeble nonblocking when the traffic is fully dynamic and satisfies Traffic Assumption B. Here, rearrangeably nonblocking means that traffic streams may change wavelengths, but not their routes, to make way for a new traffic stream.

These six OADM rings are for different traffic models, but they can all support static traffic. Hence, in Section C.3, we compare their costs $W$, $Q$, and $H$ under Traffic Assumption C. Our conclusions are given in Section C.4.

## C.2 Optical WDM Ring Architectures

### C.2.1 Fully-Optical Ring

Consider a network where traffic must be routed on a single lightpath from its source to its destination. This will require setting up lightpaths between each source and destination node between which there is any traffic. We will compute the costs for the ring assuming the static uniform traffic with parameter $g \leq c$. Then we need to set up one lightpath between each pair of nodes. This type of a network has been considered in [11].

Next, the lightpath set-up will be described using a recursive definition, that appeared in [15] and was also independently discovered in [10]. Other definitions of fully-optical rings can be found in [35].

We first consider the case when $N$ is even.

1. Start with 2 nodes on the ring (see Figure C.4.) The sole lightpath that needs be set up will require 1 wavelength.

2. (Recursive step) Let $k$ denote the number of nodes in the ring currently. While $k \leq N-2$, add 2 more nodes to the ring such that they are diametrically opposite to each other, i.e., separated by the maximum possible number of hops (see Figure C.5). The two new nodes divide the ring in half, where each half has $\frac{k}{2}$ old nodes. In one half, each old node sets up a lightpath to each new node. This

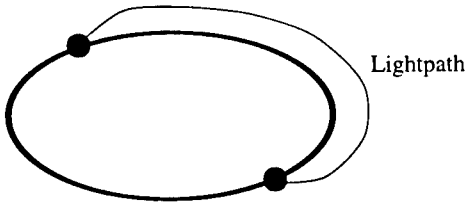



Figure C.4: Setting up a lightpath between the first two nodes.

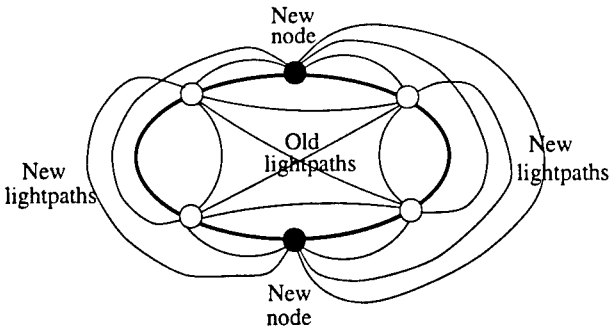



Figure C.5: Setting up the lightpaths for two new nodes.

requires one wavelength per old node since each old node can fit its two lightpaths in a wavelength (since the lightpaths use disjoint routes). Thus, a total of $\frac{k}{2}$ new wavelengths are required. The old nodes in the other half of the ring can do the same thing and use the same $\frac{k}{2}$ wavelengths. Finally, the two new nodes require an additional wavelength to set up a lightpath between them. Thus, we need to add a total of $(k/2)+1$ new wavelengths.

So the number of wavelengths needed to do the assignment is

$$W = 1 + 2 + 3 + ... + \frac{N}{2} = \frac{N^2}{8} + \frac{N}{4}.$$

For arbitrary $g$ the wavelength assignment can be done with

$$W = \left\lceil \frac{g}{c} \right\rceil \left( \frac{N^2}{8} + \frac{N}{4} \right)$$

wavelengths, where $N$ is even.

When $N$ is odd, we start the procedure above with 3 nodes and add two nodes each time. The number of wavelengths in this case can be calculated to be

$$W = \left\lceil \frac{g}{c} \right\rceil \frac{N^2 - 1}{8}.$$

Clearly, the number of transceivers required per node is given by

$$Q = \left\lceil \frac{g}{c} \right\rceil (N - 1).$$

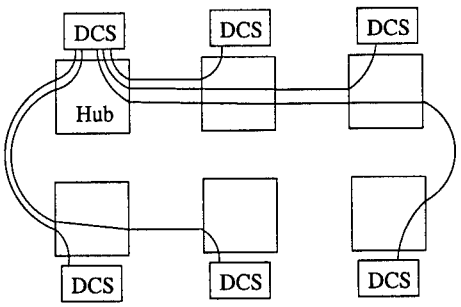



Figure C.6: A single hub network for the case when $t_A(i) = 1$ for all nodes $i$.

The maximum hop length is

$$H = \left\lfloor \frac{N}{2} \right\rfloor. \tag{C.1}$$

### C.2.2 Single-Hub Ring

The *single-hub* ring network is for Traffic Assumption A. It has a node designated as the *hub*, which will be referred to as node $h$. An example of a single-hub network is shown in Figure C.6. The hub node is chosen such that it achieves the maximum $\max_{0 \le i < N} t_A(i)$. As we shall see, this choice for the hub minimizes the required number of wavelengths. For simplicity, we will denote $\max_{0 \le i < N} t_A(i)$ by $t_{\max}$.

Each node $i$ is directly connected to the hub by $t_A(i)$ lightpaths. Thus, the logical topology of the network is a star topology. Traffic streams are routed between nodes by going through the hub. The network is wide-sense nonblocking because the DCS at the hub is wide-sense nonblocking, and for each node $i$ there are enough lightpaths provisioned to the hub to accommodate all of its traffic. Thus, we have the following theorem.

**Theorem 1** *For Traffic Assumption A, the single-hub ring is wide sense nonblocking.*

The number of wavelengths required is $\left\lceil \frac{1}{2} \sum_{i \neq h} t_A(i) \right\rceil$ because there are $\sum_{i \neq h} t_A(i)$ lightpaths, and we can fit two lightpath connections into a wavelength (the lightpaths on the same wavelength use disjoint routes along on the ring).

We have the following properties of the single hub ring:

- $W = \left\lceil \frac{1}{2} [\sum_{i=0}^{N-1} t_A(i) - t_{\max}] \right\rceil$
- $Q = 2\frac{\sum_{i=0}^{N-1} t_A(i) - t_{\max}}{N}$ since there are $\sum_{i=0}^{N-1} t_A(i) - t_{\max}$ lightpaths.
- $H = N - 1$ since lightpath routes may be forced to circumvent the ring to minimize wavelengths.

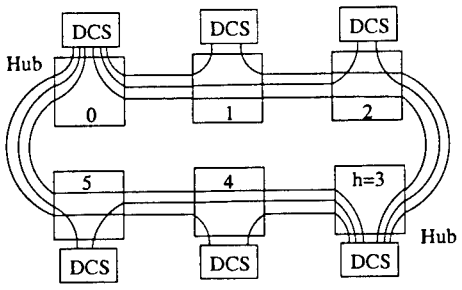



Figure C.7: A double-hub network when $t_A(i) = 2$ for all nodes $i$.

Now note that since the single-hub ring is *wide sense nonblocking*, it is also rearrangeably nonblocking. The following theorem gives a simple lower bound on the number of wavelengths required for such a OADM ring, and we give its proof for completeness. Notice that the number of wavelengths for the single-hub ring is about twice as much as the lower bound. However, in the next subsection, a rearrangeably nonblocking OADM ring is given that almost meets the lower bound.

**Theorem 2** *Consider a rearrangeably nonblocking OADM ring network under Traffic Assumption A. Suppose $N$ is even, and for each node $i = 0, 1, ..., N-1$, $t_A(i) = \tau$, where $\tau$ is some integer. Then the number of wavelengths $W$ is at least $\lceil \tau \frac{N}{4} \rceil$.*

**Proof.** Consider the case where for $i = 0, 1, ...., \frac{N}{2} - 1$, there is $c \cdot \tau$ traffic streams between the pair of nodes $i$ and $i + \frac{N}{2}$. Since each traffic stream must traverse $\frac{N}{2}$ links, there are $\frac{N}{2}$ pairs of nodes, and there are $N$ links, the average number of traffic streams going through a link must be at least

$$\frac{c \cdot \tau \cdot \frac{N}{2} \cdot \frac{N}{2}}{N} \tag{C.2}$$

which is equal to $\frac{c \cdot N \cdot \tau}{4}$. The theorem is implied. □

### C.2.3 Double-Hub Ring

The *double-hub* ring network is for Traffic Assumption A. It has two nodes that are *hubs*. An example of a double-hub ring is shown in Figure C.7. Without loss of generality, assume one of the hubs is node 0, and denote the other hub by $h$. Each node $i$ has communication connections to each hub, and the aggregate capacity to each hub is equivalent to $\frac{t_A(i)}{2}$ lightpaths. This allows node $i$ to send (and sink) up to $c\frac{t_A(i)}{2}$ traffic streams to (and from) each hub.

We will now describe how the communication connections are realized by lightpaths. We will use the following terminology and definitions. The nodes $0, 1, 2, ..., h-1$ will be

referred to as *side 1* of the ring. The rest of the nodes $h, h+1, ..., N-1$ will be referred to as *side 2* of the ring. We will also use the notation $fr\left(\frac{t_A(i)}{2}\right)$ to denote the fractional part of $\frac{t_A(i)}{2}$. Note that $fr\left(\frac{t_A(i)}{2}\right)$ is zero if $t_A(i)$ is even and $\frac{1}{2}$ if $t_A(i)$ is odd. We will refer to nodes that have odd $t_A(i)$ as *odd traffic* nodes.

We will now describe how nodes in side 1 connect to the hubs. (Note that the nodes in side 2 are connected to the hubs in a similar way.) Each node $i$ in side 1 uses $\left\lfloor \frac{t_A(i)}{2} \right\rfloor$ wavelengths to carry $\left\lfloor \frac{t_A(i)}{2} \right\rfloor$ lightpaths directly to each hub. The lightpaths are routed only using links on side 1 of the ring. Note that it is possible to use only $\left\lfloor \frac{t_A(i)}{2} \right\rfloor$ wavelengths because lightpaths going to the two hubs have disjoint routes.

Note that if $t_A(i)$ is odd then node $i$ must have an additional $\frac{1}{2}$ $(= fr\left(\frac{t_A(i)}{2}\right))$ worth of lightpath connection to each hub. These "half-a-lightpath" connections are realized by having two odd-traffic nodes share a wavelength. For example, if $u$ and $v$ are odd-traffic nodes sharing a wavelength and $u < v$ then there would be lightpaths between the pairs $(0, u)$, $(u, v)$, and $(v, h)$. Thus, if $u \neq 0$ then there would be three lightpaths, and if $u = 0$ then there would be two lightpaths. Now nodes $u$ and $v$ can use half the bandwidth of a lightpath to carry $\frac{c}{2}$ traffic streams to and from each hub.

It is straight forward to check that number of wavelengths required for side 1 of the ring is $\left\lceil \frac{1}{2} \sum_{i=0}^{h-1} t_A(i) \right\rceil$. Note that nodes 0 and $h$ each have $\left\lceil \frac{1}{2} \sum_{i=0}^{h-1} t_A(i) \right\rceil$ transceivers to terminate lightpaths on side 1. Each node $i = 1, 2, ..., h-1$ have $2\lfloor \frac{t_A(i)}{2} \rfloor$ transceivers to terminate lightpaths with a "full lightpath worth" of connection, and has $2\left\lceil fr\left(\frac{t_A(i)}{2}\right)\right\rceil$ transceivers to terminate lightpaths with "half a lightpath worth" of connection. Thus, each node $i = 1, 2, ..., h-1$ has $2\left\lceil \frac{t_A(i)}{2} \right\rceil$ transceivers. Hence, the total number of transceivers that terminate lightpaths on side 1 is $2\left\lceil \frac{1}{2} \sum_{i=0}^{h-1} t_A(i) \right\rceil + \sum_{i=1}^{h-1} 2\left\lceil \frac{t_A(i)}{2} \right\rceil$. Similar calculations can be done for side 2. Thus, we have

- $W = \max\left\{ \left\lceil \frac{1}{2} \sum_{i=0}^{h-1} t_A(i) \right\rceil, \left\lceil \frac{1}{2} \sum_{i=h}^{N-1} t_A(i) \right\rceil \right\}$,
- $Q = \frac{1}{N}\left(2\left\lceil \frac{1}{2} \sum_{i=0}^{h} t_A(i) \right\rceil + 2\left\lceil \frac{1}{2} \sum_{i=h+1}^{N-1} t_A(i) \right\rceil + \sum_{i=1}^{h-1} 2\left\lceil \frac{t_A(i)}{2} \right\rceil + \sum_{i=h+1}^{N-1} 2\left\lceil \frac{t_A(i)}{2} \right\rceil\right)$
- $H = \max\{h, N-h\}$

To compare $W$ with the simple lower bound in Theorem 2, let $N$ be even, $h = N/2$, and for all nodes $i$, $t_A(i) = \tau$, where $\tau$ is some integer. Then $W = \lceil \tau N/4 \rceil$, which equals the lower bound.

**Theorem 3** *Consider Traffic Assumption A and the double-hub ring network. Suppose $c$ is even and, for each node $i$, $c \cdot t_A(i)$ is divisible by four. Then the double-hub ring network is rearrangeably nonblocking.*

**Proof.** The double-hub ring can be viewed as a switching network where lower-speed traffic streams are routed between nodes via hub nodes. Note that the traffic streams are full duplex (i.e., bidirectional) so they do not have distinct source and destination nodes typically used to define connections in switching networks. We will artificially give each traffic stream a *direction*, so that it will have a source and destination. Note that the directions are used for routing purposes only, and the traffic streams are still full duplex. Also note that the directions for traffic streams may change over time which may be necessary for rerouting.

We can assume that any collection of traffic streams, can be directed so that at each node $i$, at most $c\frac{t_A(i)}{2}$ streams are directed into it or out of it. This assignment can be done as follows. Since each node $i$ has an even value for $c \cdot t_A(i)$, we may assume that each node $i$ terminates exactly $c \cdot t_A(i)$ traffic streams. Otherwise, the assumption can be made true by greedily adding dummy streams, where streams that have both their ends terminating at a single node are allowed. Since there is an even number of traffic streams incident to any node, we can find an *Eulerian walk* [25] where the streams are treated as *edges* in a *multigraph* (i.e., a graph that can have multiple edges between pairs of nodes), and *self-loop* edges (i.e., edges from a vertex to itself) are possible. Recall that such a walk is a tour that visits each edge exactly once and then returns to the starting node. The traversal of such a tour gives directions to the streams such that at each node $i$, exactly $c\frac{t_A(i)}{2}$ (real or dummy) streams are directed into and out of it.

With the traffic streams directed, the double-hub ring can be viewed as emulating a three stage switching network, as shown in Figure C.8, that supports directed traffic streams. The first stage has $N$ vertices denoted by $s_0, s_1, ..., s_{N-1}$, where $s_i$ represents node $i$ in the ring network. The second stage has two vertices representing the two hubs. The third stage has $N$ vertices denoted by $d_0, d_1, ..., d_{N-1}$, where $d_i$ also represents node $i$ in the ring network. Hence, node $i$ in the ring is represented by two vertices $s_i$ and $d_i$ in the three stage switching network. (It will be shown shortly that if $i$ is a hub then it is also represented by a vertex in the second stage of the three stage switching network.)

Each vertex $s_i$ in the first stage has $c\frac{t_A(i)}{2}$ input links which represents the fact that node $i$ in the ring can source $c\frac{t_A(i)}{2}$ directed traffic streams. Similarly, each vertex $d_i$ in the third stage has $c\frac{t_A(i)}{2}$ output links which represents the fact that node $i$ in the ring can be the destination of $c\frac{t_A(i)}{2}$ directed traffic streams.

Each vertex $s_i$ in the first stage has $c\frac{t_A(i)}{4}$ links to each vertex in the second stage, and each vertex $d_i$ in the third stage has $c\frac{t_A(i)}{4}$ links from each vertex in the second stage. Thus, vertices $s_i$ and $d_i$ together have $c\frac{t_A(i)}{2}$ links to each hub. These links represent the fact that node $i$ in the ring network can have $c\frac{t_A(i)}{2}$ traffic streams to each hub.

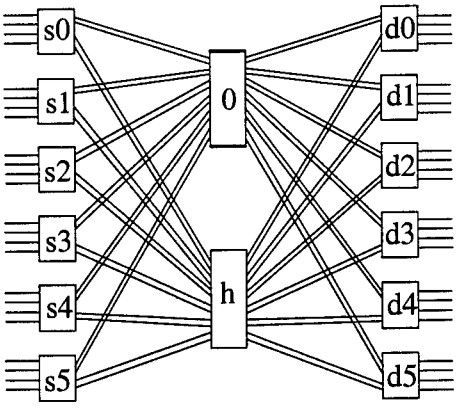



Figure C.8: A three stage switch for $N = 6$, $c = 4$, and $t_A(i) = 2$ for all nodes $i$.

The three stage switching network is rearrangeably nonblocking. This can be shown by first transforming it into a three stage Clos network (see [19] for a description of a Clos network). In particular, each vertex $s_i$ in the first stage is transformed into $c\frac{t_A(i)}{2}$ vertices, each having two input links and one link to each second stage vertex. Similarly, each vertex $d_i$ in the third stage is transformed into $c\frac{t_A(i)}{2}$ vertices, each having two output links, and one link from each second stage vertex. The Clos network is rearrangeably nonblocking because there are two input links at each first stage vertex, two output links at each third stage vertex, and two vertices in the second stage [33, 9]. The original three stage network is rearrangeably nonblocking because it can emulate the Clos network. Hence, the double hub ring is rearrangeably nonblocking. □

### C.2.4 Point-to-Point WDM Ring

We will present the costs of the PPWDM ring network assuming Traffic Assumption B. The network has $W = L$. Therefore, the network is wide sense nonblocking. For the ring, obviously, the number of transceivers per node is

$$Q = 2W, \tag{C.3}$$

and the maximum hop length is

$$H = 1. \tag{C.4}$$

### C.2.5 Hierarchical Ring

In this section we will describe the *hierarchical ring*. Throughout the section we will assume Traffic Assumption B. To simplify the discussion, we will also assume that for each node $i$, $t_B(i) = \tau$, where $\tau$ is some integer parameter.

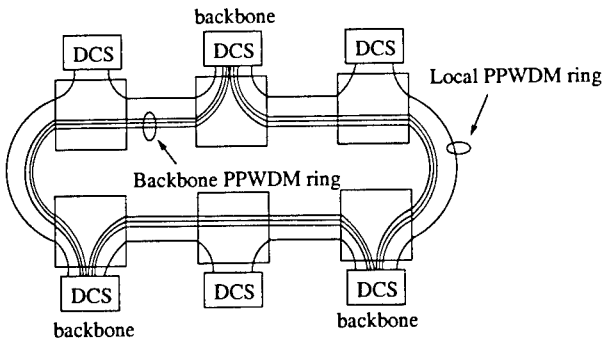



Figure C.9: A hierarchical ring with parameter $\alpha = 2$, $\tau = 1$, and $L = 3$.

The network is has an integer $\alpha > 0$, and has a total of $W = L + \tau \cdot (\alpha - 1)$ wavelengths. $L$ of the wavelengths are referred to as *backbone* wavelengths, and the other $\tau(\alpha - 1)$ wavelengths are referred to as *access* wavelengths. The nodes of the ring are also classified into *access* or *backbone* types. Note that the backbone nodes are 0, $\alpha$, $2\alpha$,..., $(\lceil \frac{N}{\alpha} \rceil - 1)\alpha$, while the other nodes are access nodes. Thus, the nodes are arranged such that there are at most $\alpha - 1$ access nodes between any consecutive pair of backbone nodes. The backbone wavelengths are used to form a PPWDM ring among the backbone nodes. In other words, lightpaths are formed between consecutive backbone nodes using the backbone wavelengths. We will refer to this as the *backbone PPWDM ring.* Note that lightpaths at the backbone wavelengths have at most $\alpha$ hops.

The access wavelengths are used to form a PPWDM ring among all nodes, both access and backbone types. In other words, lightpaths are formed between consecutive nodes using access wavelengths. We will refer to this as the *access PPWDM ring.* The lightpaths at the access wavelengths have one hop.

Figure C.9 shows an example hierarchical ring. It is straight forward to show the following properties of the hierarchical ring.

- $W = L + (\alpha - 1)\tau$.
- $Q = 2(\alpha - 1)\tau + 2\frac{L}{N}\lceil \frac{N}{\alpha} \rceil$.
- $H = \alpha$.

Note that the value of $\alpha$ can be chosen to minimize $Q$. For large $N$, the optimal value of $\alpha$ is approximately $\sqrt{L/\tau}$.

The lightpath assignment algorithm for the ring network is as follows. Consider an arriving traffic stream with its route. Suppose we follow the route in the clockwise direction around the ring. Let $u_0$ and $v_1$ be the first and last nodes of the route. Let $v_0'$ and

$v_1'$ be the first and last backbone nodes, respectively, along the route. The traffic stream from $v_0$ to $v_0'$ and then from $v_1'$ to $v_1$ will be assigned lightpaths from the access PPWDM ring. Between nodes $v_0'$ and $v_1'$ the traffic stream will be assigned lightpaths from the backbone PPWDM ring. Thus, the access PPWDM ring serves as an access network to the backbone PPWDM ring, and the backbone PPWDM ring transports traffic streams across the network. The traffic stream will not be blocked because enough wavelengths have been provisioned. In particular, the $L$ wavelengths of the backbone PPWDM ring are sufficient to transport traffic across the network. Also, the $(\alpha - 1)\tau$ wavelengths of the access PPWDM ring are sufficient for traffic streams to access neighboring backbone nodes because between backbone nodes there are at most $(\alpha - 1)$ access nodes and each of then can terminate at most $c \cdot \tau$ traffic streams from the clockwise or counter-clockwise direction along the ring. Thus, we have the following result.

**Theorem 4** *Consider Traffic Assumption B. The hierarchical ring network is wide sense nonblocking.*

### C.2.6 Incremental Ring

In this section we will consider the *incremental ring network.* Throughout the section we will assume Traffic Assumption B and incremental traffic. The network has $W = L$ wavelengths which is the same number as for the PPWDM ring. We will first describe the network architecture, and then show that it is nonblocking for incremental traffic. At the end of the section, we will present a strategy to configure the architecture to minimize transceiver cost $Q$.

The incremental ring network is organized with respect to pieces of it called *subnets.* A subnet $s$ is composed of a sequence of links (and nodes) along the ring in the clockwise direction. This sequence will be referred to as its *segment.* The subnet's *length* is the length of its segment. The subnet's *end nodes* are the nodes at the end of its segment. The rest of its nodes are referred to as the *internal nodes* and denoted by $I(s)$. Figure C.10 shows a segment and its end and internal nodes.

A subnet $s$ is also composed of a collection of wavelengths and lightpaths. The wavelengths are $w_0, w_1, ...., w_{r(s)-1}$, where $r(s)$ denotes the number of wavelengths for $s$. The lightpaths only use the wavelengths and links of $s$.

The incremental ring architecture is organized as a *tree* of subnets. The *root* subnet has a segment that includes all the links of the ring. The segment starts and ends at some node, which we refer to as the *root node.* The number of wavelengths for this subnet is $W$, i.e, $r(root) = W$.

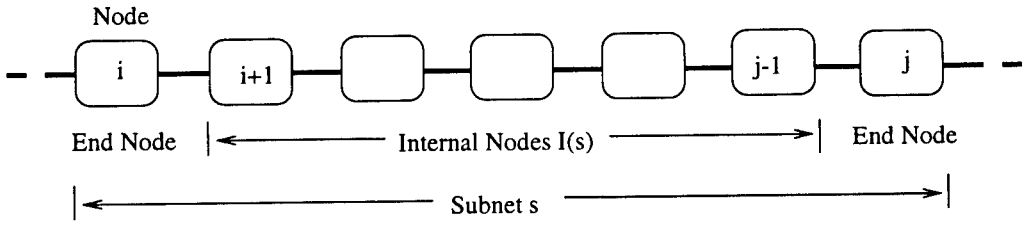



Figure C.10: The segment of subnet $s$.

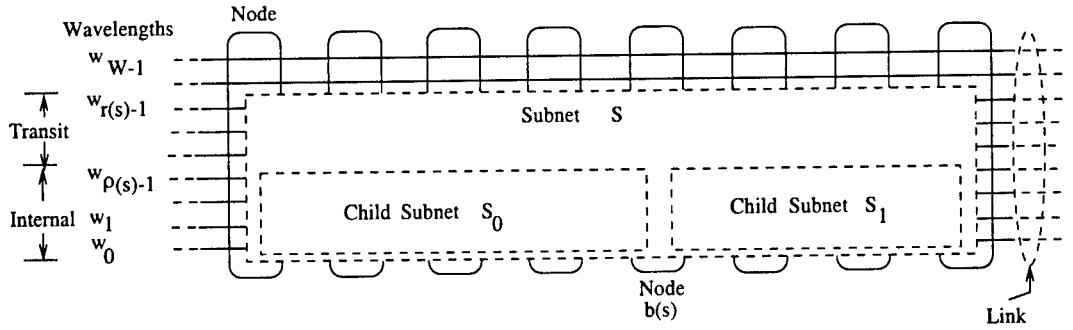



Figure C.11: A parent subnet $s$ composed of its two children subnets $s_0$ and $s_1$.

If a subnet $s$ has length greater than one (i.e., it has at least one internal node) then it is referred to as a *parent* subnet. Note that the *root* subnet is a parent if $N > 1$. A parent subnet has two *children* subnets, say $s_0$ and $s_1$. The segments of the children are defined by bisecting the segment of $s$ at some internal node denoted by $b(s) \in I(s)$. The resulting subsegments are the segments for $s_0$ and $s_1$. The number of wavelengths assigned to each of the children is $r(s_0) = r(s_1) = \rho(s)$, where $\rho(s)$ is defined to be

$$\rho(s) = \min\{r(s), \sum_{i \in I(s)} t_B(i)\}. \tag{C.5}$$

Note that $\rho(s)$ is equal to

$$\min\{W, \sum_{i \in I(s)} t_B(i)\} \tag{C.6}$$

since $r(root) = W$ and for an arbitrary parent subnet $s'$ and its child $s^*$, $\sum_{i \in I(s')} t_B(i) \geq \sum_{i \in I(s^*)} t_B(i)$. Figure C.11 shows an example subnet $s$, its bisecting node $b(s)$, and two children $s_0$ and $s_1$. As shown in the figure, the $r(s)$ wavelengths of $s$ are categorized into two types for $s$: *internal* and *transit*. The internal wavelengths of $s$ are $\{w_0, w_1, ..., w_{\rho(s)-1}\}$, and the transit wavelengths of $s$ are $\{w_{\rho(s)}, w_{\rho(s)+1}, ..., w_{r(s)-1}\}$. Thus, the $\rho(s)$ internal wavelengths of $s$ are the wavelengths of its children. This categorization of wavelengths will be used later to define where lightpaths are placed in the subnet, and how lightpaths are assigned to traffic streams.

If a subnet $s$ has only one link then it has no children and is called a *leaf* subnet. Leaf subnets only have transit wavelengths. Now note that the collection of leaf and parent subnets form a tree of subnets, where the root and leaf subnets are the root and leaves, respectively, of the tree. Also note that a subnet can be both a child (of some subnet) and a parent (of two other subnets).

Next, we define how transceivers are placed at each node, which will in turn define the placement of lightpaths. First, the root node has $2W$ transceivers terminating each wavelength on both of its links. For all other nodes, the placement is as follows. A node must be a bisecting node $b(s)$ for some parent subnet $s$. The node $b(s)$ has $2\rho(s)$ transceivers, which terminate wavelengths $w_0, ...w_{\rho(s)-1}$ on each of the node's links. Thus, for subnet $s$, its end and bisecting nodes terminate lightpaths on its internal wavelengths, while *only* its end nodes terminate lightpaths on its transit wavelengths (refer to Figure C.11). Thus, lightpaths on its transit wavelengths directly connect its end nodes, and cannot be used for traffic streams that terminate at its internal nodes.

To complete the description of the incremental ring architecture, we define its *lightpath assignment algorithm* (LAA). The LAA takes an incoming traffic stream with a route, and finds a contiguous sequence of lightpaths along the route that can carry the stream. To define the LAA for the ring, we will define LAAs for the subnets.

The LAA for a subnet takes a route and assigns a contiguous sequence of its lightpaths along the route that have spare capacity. Thus, a traffic stream following the route can be carried by the lightpaths. The route is assumed to be entirely in the segment of the subnet. Thus, the route may actually be part of some larger traffic stream route that goes through the subnet. There are two kinds of routes: *internal* and *transit*. An internal route ends at at least one internal node of the subnet, while a transit route does not, i.e., it ends at the end nodes.

We will describe the LAA for a subnet $s$ by first assuming that $s$ is a parent subnet, as shown in Figure C.11. How the LAA assigns lightpaths to a route depends on whether the route is internal or transit. If the route is internal then it is first split at the bisecting node $b(s)$. (Of course, it is not split if it does not pass through $b(s)$.) The resulting subroutes fit into the segments of the children subnets of $s$. The LAAs of the children are used to assign lightpaths to the subroutes. These assignments combine to provide the assignment for the original route. However, if the route is transit, then the LAA will first attempt to assign a lightpath on a transit wavelength of $s$ to the route. If all such lightpaths have no spare capacity then the LAA assigns lightpaths to the route as if it were an internal route, i.e., it uses the LAAs of the children subnets.

The LAA for a leaf subnet $s$ is simpler because all routes and wavelengths are transit.

A route is assigned to a lightpath of $s$ with spare capacity.

We have completed the definition of the LAAs for subnets. The LAA for the root subnet is basically the LAA for the incremental ring. In particular, the LAA for the incremental ring will assign lightpaths to an incoming traffic stream as follows. It first splits the route of the stream at the root node (if the route passes through the root). The resulting subroutes are entirely in the root subnet and can be assigned lightpaths by the subnet's LAA. This lightpath assignment is an assignment for the traffic stream since the lightpaths that meet at the root node can be crossconnected.

Next we will prove the following property.

**Theorem 5** *Consider an incremental ring network under Traffic Assumption B and incremental traffic. Then the incremental ring network is wide sense nonblocking.*

The theorem is implied by the next lemma, which uses the following definitions. Consider a subnet $s$. The subnet is said to have *spare capacity* in a link if one or more of its wavelengths have spare capacity in the link. The subnet is said to be *wide sense nonblocking* if given a route, the LAA of the subnet will find a lightpath assignment for the route as long as the subnet has spare capacity in each link along the route. Note that having the spare capacity does not preclude a valid lightpath assignment because in order for a set of lightpaths to carry a traffic stream they must terminate at the ends of the route, and at common intermediate nodes. Note that the next lemma implies Theorem 5 because it is applicable to the root subnet.

**Lemma 1** *Consider an incremental ring network under Traffic Assumption B and incremental traffic. Then each subnet $s$ of the ring is wide sense nonblocking.*

*Proof.* To prove the lemma for leaf subnets is trivial since the subnet has only one link. Thus, we will only consider parent subnets. The proof of the lemma is by induction. Consider a parent subnet $s$ and its two children subnets $s_0$ and $s_1$. We will assume that the children subnets are nonblocking, and proceed to show that $s$ is also nonblocking.

Let us see how subnet $s$ assigns lightpaths to traffic streams over time. Initially, there are no traffic streams so all lightpaths in the subnet are empty. As traffic streams arrive they are assigned lightpaths so that internal traffic (i.e., streams that terminate at internal nodes of $s$) are assigned to lightpaths with internal wavelengths, and transit traffic (i.e., streams that do not terminate at internal nodes of $s$) are assigned to lightpaths with transit wavelengths.

Note that this can continue while transit traffic does not completely fill lightpaths with transit wavelengths. This is partly due to the traffic model which assumes that each

internal node $i \in I(s)$ can terminate at most $c \cdot t_B(i)$ traffic streams from the clockwise or counter-clockwise direction. This implies that for any link of the subnet, the number of internal traffic streams that go over it is at most $c \cdot \rho(s)$. The internal traffic can be accommodated by the lightpaths with internal wavelengths because

- children subnets are assumed to be nonblocking and occupy the $\rho(s)$ internal wavelengths; and
- bisecting node $b(s)$ (which is between the children subnets) terminates all the internal wavelengths with transceivers, and can therefore crossconnect the spare capacity in the two children subnets.

Thus, while transit streams do not completely fill the transit wavelengths, internal traffic streams will be accommodated by lightpaths at internal wavelengths. Hence, there will be no blocking.

Now at some point the lightpaths with transit wavelengths may become completely filled. Then the only spare capacity left in the subnet $s$ is in its internal wavelengths. Arriving traffic streams will not be blocked from this spare capacity because (i) the spare capacity resides in the nonblocking children subnets, and (ii) the bisecting node $b(s)$ can crossconnect the spare capacity in the two children subnets. □

Finally, we will describe a strategy to place transceivers (and lightpaths) to minimize the transceiver cost. Notice that the placement of transceivers is dependent on the choice of root and bisecting nodes. Thus, the strategy is to find an optimal choice of these nodes. Before presenting the strategy, we give some definitions.

- Let $\sigma(i,k)$ denote a segment of the ring that starts at node $i$ and goes clockwise over $k$ links.
- Let $I'(i,k)$ denote the *internal* nodes of $\sigma(i,k)$, i.e., $I'(i,k) = \{(i+j) \bmod N : 0 < j < k\}$.
- Consider all possible ways of configuring the incremental ring, i.e., all possible ways of choosing the root and bisecting nodes. Of these, consider all configurations that have a subnet with segment $\sigma(i,k)$. Let $\mathcal{C}(i,k)$ denote these configurations of the incremental ring.
- For each configuration $\gamma \in \mathcal{C}(i,j)$, let $q_\gamma(i,k)$ be the number of transceivers placed at the internal nodes $I'(i,k)$ assuming the incremental ring is configured as $\gamma$. Let $q(i,k) = \min_{\gamma \in \mathcal{C}(i,j)} q_\gamma(i,k)$. Thus, $q(i,k)$ is the minimum number of transceivers at the internal nodes $I'(i,k)$ assuming that $\sigma(i,k)$ is the segment of a subnet.

Notice that for each node $i$, $2W + q(i, N)$ is the minimum number of transceivers in the incremental ring assuming $i$ is the root node because there are $2W$ transceivers at $i$ and a minimum of $q(i, N)$ transceivers at all other nodes. Thus, the minimum number of transceivers for the incremental ring is $2W + \min_i q(i, N)$.

We now describe the strategy to determine the root and bisecting nodes. It consists of two steps. The first step is to compute the values for $q(i, k)$ for all nodes $i$ and all $k = 1, 2, ..., N$ by using the equation

$$q(i,k) = \begin{cases} 0, & \text{if } k = 1 \\ \min_{i<j<k} \phi(i,j,k) & \text{if } k > 1 \end{cases} \tag{C.7}$$

where

$$\phi(i,j,k) = q(i,j) + q((i+j) \bmod N, k-j) + 2\min\{W, \sum_{n \in I'(i,k)} t_B(n)\}.$$

The equation is true for $k = 1$ because $\sigma(i, 1)$ corresponds to a leaf subnet, which has no internal nodes. The equation is also true for $k > 1$ because (i) the minimum is over all bisecting nodes $(i + j)$ mod $N$ for a subnet with segment $\sigma(i, k)$, (ii) the terms $q(i, j)$ and $q(i+j, k-j)$ are the minimum number of transceivers at internal nodes of the two children of the subnet, and (iii) the term $2\min\{W, \sum_{n \in I'(i,k)} t_B(n)\}$ is the number of transceivers at the bisecting node which is twice Expression (C.6). The values for $\{q(\cdot, \cdot)\}$ are determined by first computing $q(i, 2)$ for all nodes $i$. Then computing $q(i, 3)$ for all nodes $i$, and so forth.

The second step is to determine the root and bisecting nodes that minimize the number of transceivers in the network from the values of $\{q(\cdot, \cdot)\}$. This is straightforward. In particular, the node $i$ that minimizes $\min_i q(i, N)$ is the optimal root node. The node $(i + j)$ mod $N$ that minimizes the right hand side of (C.7) is the optimal bisecting node for the corresponding subnet. It is straightforward to show that this two step strategy has time complexity $O(N^3)$.

## C.3 Comparisons

In this section we will compare the six OADM ring networks in the previous section. Note that the networks operate under different traffic models. Table C.1 lists the networks with their traffic models and switching capabilities. To compare their costs, we will use the uniform static traffic model with parameter $g$ because all six networks can operate under this model. Recall that the traffic has $g$ traffic streams between every pair of nodes,

| Network | Traffic Type | Switching Capability |
|---|---|---|
| Fully Optical | static | not applicable |
| Single-Hub | dynamic, traffic assumption A | wide-sense nonblocking |
| Double-Hub | dynamic, traffic assumption A | rearrangeably nonblocking |
| PPWDM | dynamic, traffic assumption B | wide-sense nonblocking |
| Hierarchical | dynamic, traffic assumption B | wide-sense nonblocking |
| Incremental | incremental, traffic assumption B | wide-sense nonblocking |

Table C.1: A comparison of the traffic models and switching capabilities of the six OADM Rings.

i.e.,

$$T(i,j) = \begin{cases} g/c & \text{if } i \neq j \\ 0 & \text{otherwise} \end{cases} \tag{C.8}$$

To simplify the comparison we will also assume that $N$ is a power of two.

Note that the costs of the fully optical ring have already been presented for the static uniform traffic. We will proceed to present the costs of the other five of the OADM rings in the following order: single-hub, double-hub, PPWDM, hierarchical, and incremental.

For the single-hub and double-hub rings, the costs $W$ and $Q$ are a function of $(t_A(i) : i = 0, 1, ..., N-1)$. For the static uniform traffic, $t_A(i) = u_A(g)$ for all nodes $i$, where

$$u_A(g) = \left\lceil \frac{g(N-1)}{c} \right\rceil. \tag{C.9}$$

because each node terminates $g(N-1)$ traffic streams. Thus, we have the following costs.

**Single-Hub Ring:**

- $W = \left\lceil \frac{u_A(g)\cdot(N-1)}{2} \right\rceil$
- $Q = 2u_A(g)(1 - \frac{1}{N})$
- $H = \frac{N}{2}$, since we can arrange the lightpaths to take shortest hop paths.

**Double-Hub Ring:** Assuming the hubs are 0 and $\frac{N}{2}$,

- $W = \left\lceil u_A(g)\frac{N}{4} \right\rceil$
- $Q = \frac{1}{N}\left(4\left\lceil \frac{N}{4}u_A(g) \right\rceil + (N-2)2\left\lceil \frac{u_A(g)}{2} \right\rceil\right)$
- $H = \frac{N}{2}$

For the PPWDM, hierarchical, and incremental rings, the traffic streams should satisfy Traffic Assumption B. We will assume that the pre-routes of the traffic streams are shortest hop paths. Next we determine appropriate values for $\{t_B(i) : i = 0, 1, ..., N-1\}$ and $L$.

Note that since $N$ is even, nodes at opposite ends of the ring (e.g., arbitrary node $i$ and node $\left(i + \frac{N}{2}\right) \bmod N$) have two shortest paths, each on opposite sides of the ring. We will assume that the traffic streams between them are split as evenly as possible among the two paths. Thus, one path will have $\lceil g/2 \rceil$ streams, while the other will have $\lfloor g/2 \rfloor$ streams. Then the node requires $u_B(g)$ transceivers on each of its links just to terminate its traffic streams, where

$$u_B(g) = \left\lceil \frac{1}{c} \left( \frac{g \cdot (N-2)}{2} + \lceil g/2 \rceil \right) \right\rceil. \tag{C.10}$$

Therefore, we set $t_B(i) = u_B(g)$ for all nodes $i$.

Also note that the average number of hops of a traffic stream is $\mathcal{H}_{Avg} = \frac{N+1}{4} + \frac{1}{4(N-1)}$. (Note that for arbitrary $N$

$$\mathcal{H}_{Avg} = \begin{cases} \frac{N+1}{4} & N \text{ odd}, \\ \frac{N+1}{4} + \frac{1}{4(N-1)} & N \text{ even.)} \end{cases}$$

Thus, the average number of traffic streams going through a link is at least

$$\begin{aligned} \ell(g) &= \frac{\mathcal{H}_{Avg} \times \text{ Total traffic}}{\text{Number of links}} = \frac{\mathcal{H}_{Avg} \times \frac{1}{2} \sum_i \sum_j c \cdot T(i,j)}{N} \\ &= g \cdot (N-1) \left( \frac{N+1}{8} + \frac{1}{8(N-1)} \right) \end{aligned}$$

Therefore, we can set

$$L = \left\lceil \frac{\ell(g)}{c} \right\rceil \tag{C.11}$$

Then we have the following costs.

**PPWDM Ring:**

- $W = \left\lceil \frac{\ell(g)}{c} \right\rceil$
- $Q = 2 \left\lceil \frac{\ell(g)}{c} \right\rceil$
- $H = 1$

**Hierarchical Ring:**

- $W = \left\lceil \frac{\ell(g)}{c} \right\rceil + (\alpha - 1)u_B(g)$
- $Q = 2(\alpha - 1)u_B(g) + \frac{2}{N}\left\lceil \frac{\ell(g)}{c} \right\rceil \left\lceil \frac{N}{\alpha} \right\rceil$
- $H = \alpha$

**Incremental Ring:**

- $W = \left\lceil \frac{\ell(g)}{c} \right\rceil$

To determine the other costs $Q$ and $H$, we assume that the the bisecting nodes for each subnet is exactly in the middle because this will minimize $Q$. Thus, all subnets have *length* (i.e., length of its segment) that are powers of two. If a subnet's length is $2^i$ (for some $i$) then its bisecting node terminates $\min\{W, u_B(g)(2^i - 1)\}$ wavelengths. Let $J$ be the largest value such that $W > u_B(g)(2^J - 1)$. Then a lightpath can pass through a subnet of length $2^J$ without going through an intermediate transceiver. However, a lightpath cannot make such a pass through a subnet of length $2^{J+1}$ because the bisecting node terminates all $W$ wavelengths. Thus,

- $H = 2^J$, and we also have the simple upper bound $H \leq \left\lfloor \frac{W}{u_B(g)} + 1 \right\rfloor$

Finally, note that

- $Q = \frac{1}{N}\left(2W + \sum_{i=0}^{\log_2 N - 1} 2^i \cdot 2\min\{W, u_B(g)\left(\frac{N}{2^i} - 1\right)\}\right)$,

since the root node terminates all $W$ wavelengths, and there are $2^i$ subnets of length $\frac{N}{2^i}$. We also have a simple upper bound for $Q$:

$$Q \leq \frac{1}{N}\left(2W + \sum_{i=0}^{\log_2 N - 1} 2^i \cdot 2u_B(g)\frac{N}{2_i}\right) \leq 2\left(\frac{W}{N} + u_B(g)\log_2 N\right) \qquad \text{(C.12)}$$

To simplify the comparison of the six OADM ring networks, we provide Table C.2 which has *asymptotic relative costs* assuming $N$ is large. The *relative costs* are a ratios $W/W_{\min}$, $Q/Q_{\min}$, and $H/H_{\min}$, where $W_{\min} = \lceil \ell(g)/c \rceil$, $Q_{\min} = \tau_A(g)$, and $H_{\min} = 1$. Note that $W_{\min}$, $Q_{\min}$, and $H_{\min}$ are simple lower bounds for the costs $W$, $Q$, and $H$, respectively. The asymptotic relative costs given in the table are approximate because small terms are ignored for large $N$. For example, for the hierarchical ring, $W/W_{\min}$ is larger than one, but the entry in the table is 1 because $\lim_{N\to\infty} W/W_{\min} = 1$. Also note that for the hierarchical ring, we assume that $\alpha$ is approximately $\sqrt{\frac{N}{4}}$ ($\approx \sqrt{\frac{L}{\tau_B(g)}}$), which minimizes $Q$.

Based upon Table C.2 we draw the following conclusions:

| | Asymptotic Relative Costs | | |
|---|---|---|---|
| | $W/W_{\min}$ | $Q/Q_{\min}$ | $H/H_{\min}$, where $H$ |
| Fully Optical | $\frac{\lceil g/c \rceil}{g/c}$ | $\frac{\lceil g/c \rceil}{g/c}$ | $\frac{N}{2}$ |
| Single-Hub | 4 | 2 | $\frac{N}{2}$ |
| Double-Hub | 2 | 2 | $\frac{N}{2}$ |
| PPWDM | 1 | $\frac{1}{4}N$ | 1 |
| Hierarchical ($\alpha = \sqrt{N/4}$) | 1 | $N^{0.5}$ | $\frac{1}{4}N^{0.5}$ |
| Incremental | 1 | $2\log_2 N$ | $\frac{N}{4}$ |

Table C.2: Asymptotic relative costs for different OADM ring networks assuming the static uniform traffic with parameter $g$, and $N$ is large and a power of two.

- If wavelengths are plentiful then the single-hub ring is a good choice since it has low transceiver cost and can support dynamic traffic. The double-hub ring is a good choice if the traffic is static (and not necessarily uniform), since it requires only half the number of wavelengths and has about the same transceiver cost.

- If wavelengths are precious then the PPWDM, hierarchical, and incremental rings are reasonable choices for OADM ring networks since they use minimal wavelengths. The PPWDM ring provides the most efficient use of wavelengths for dynamic traffic. If there are some spare wavelengths then the hierarchical ring can potentially reduce the transceiver cost. If the traffic is static (and not necessarily uniform) or incremental then the incremental ring is a good choice since it minimizes wavelengths and has low transceiver cost.

An interesting point is that the fully-optical network has the smallest transceiver cost in the range $\frac{c}{2} < g \leq c$. For this range, each pair of nodes has at least half a lightpath worth of traffic between them. For smaller values of $g$, the single-hub and double-hub rings have lower transceiver cost.

Note that Table C.2 is based on the unrealistic assumption that $N$ is very large. However, Figures C.12 and C.13 show $W/W_{\min}$ and $Q/Q_{\min}$ values, respectively, for the more realistic value of $N = 8$. In the figures, $c = 16$ (e.g., traffic stream = OC-3, lightpath = OC-48), $g$ ranges from 1 to $c = 16$, and it is assumed that the hierarchical ring has $\alpha = 2$. Figure C.12 shows that the PPWDM and incremental rings have the optimal $W$. Figure C.13 shows that minimum $Q$ is attained by different networks for different values of $g$. For $g = 1$ and 2, the single-hub ring has the smallest $Q$. For $g > c/2$, the fully optical ring has the smallest $Q$. Surprisingly, for $2 < g \leq c/2$, the incremental ring has the smallest $Q$ Thus, for small to moderate values of $g$ (which is where traffic grooming

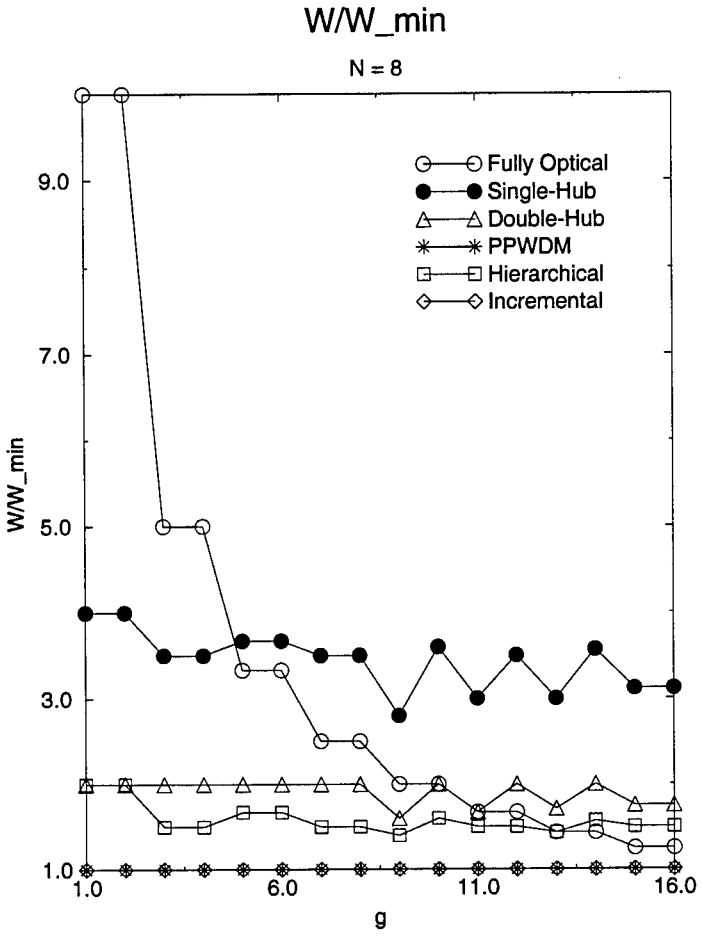



Figure C.12: Relative cost $W/W_{\min}$ for $N = 8$, $c = 16$, and $g = 1, 2, ..., 16$.

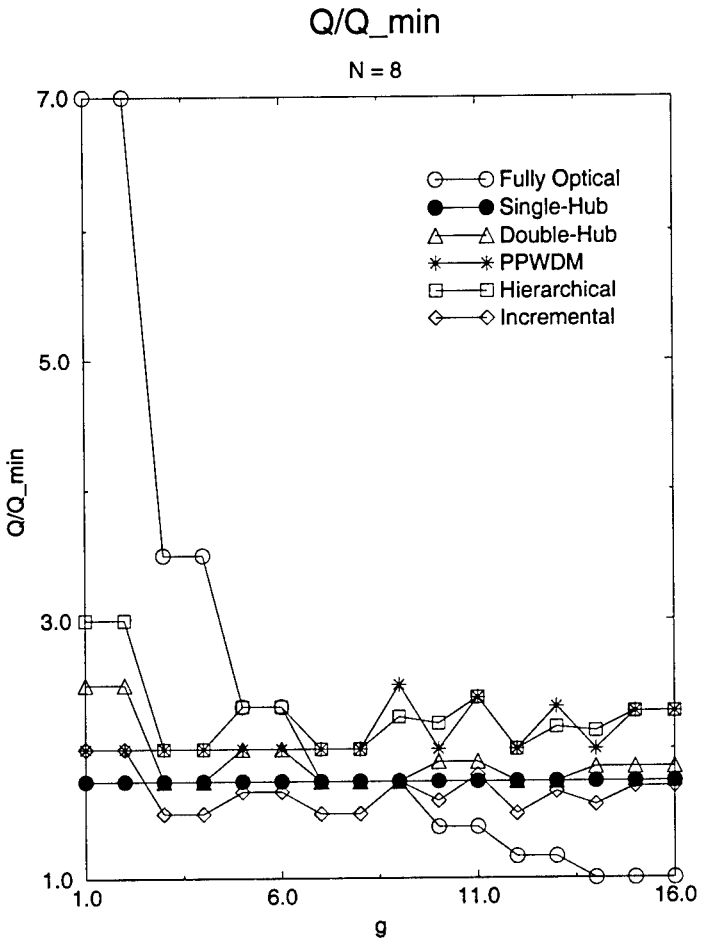



Figure C.13: Relative cost $Q/Q_{\min}$ for $N = 8$, $c = 16$, and $g = 1, 2, ..., 16$.

is most interesting), the incremental ring attains the minimum or is near minimum for both $Q$ and $W$. In fact, its improvement in $Q$ over the PPWDM ring can be significant. For example, in the case $g = 4$, the incremental ring has a $Q$ that is 25% smaller than the one for the PPWDM ring. To translate this into dollars, suppose $W = 32$ and the cost of a "transceiver" is \$75K, which includes half-an-ADM (\$50K), a transponder (\$20K), and a WDM cost per wavelength per terminal (\$10K). Then a node in the PPWDM ring will cost $2 \times W \times \$75K = \$4800K$, and a 25% cost savings translates to $\$1200K$.

Notice that the hierarchical ring has higher cost than the PPWDM ring for all values of $g$. This is due to the $N$ being a relatively small value of 8. Small $N$ means small average hop lengths for traffic streams (average hop length of approximately 2.3), which in turn implies low amounts of transit traffic. For larger $N$, the hierarchical ring will have better cost than the PPWDM ring. Figures C.14 and C.15 show $W/W_{\text{min}}$ and $Q/Q_{\text{min}}$ values, respectively, for $N = 16$. Here, $c = 16$ and $\alpha = 2$ as before. Now the transceiver cost $Q$ for the hierarchical ring is smaller by about 25% than for the PPWDM ring. The $Q$ for the incremental ring is 44% smaller than for the PPWDM ring for even values of $g$. Note if a PPWDM node costs \$4800K then a 44% savings translates to \$2112K.

## C.4 Conclusions

We have proposed and analyzed a number of OADM ring networks. At one extreme is the single-hub ring that requires large amounts of bandwidth (wavelengths) but has small transceiver cost. At the other extreme is the PPWDM ring that requires minimal bandwidth (wavelengths) but has maximum transceiver cost. In the middle we have the hierarchical ring that provides a trade-off between numbers of wavelengths and transceiver costs. Also in the middle, we have the double-hub and incremental rings. These last two do not support fully dynamic traffic, but seem to be reasonable solutions for static nonuniform traffic, and in the case of the incremental ring can support incremental traffic.

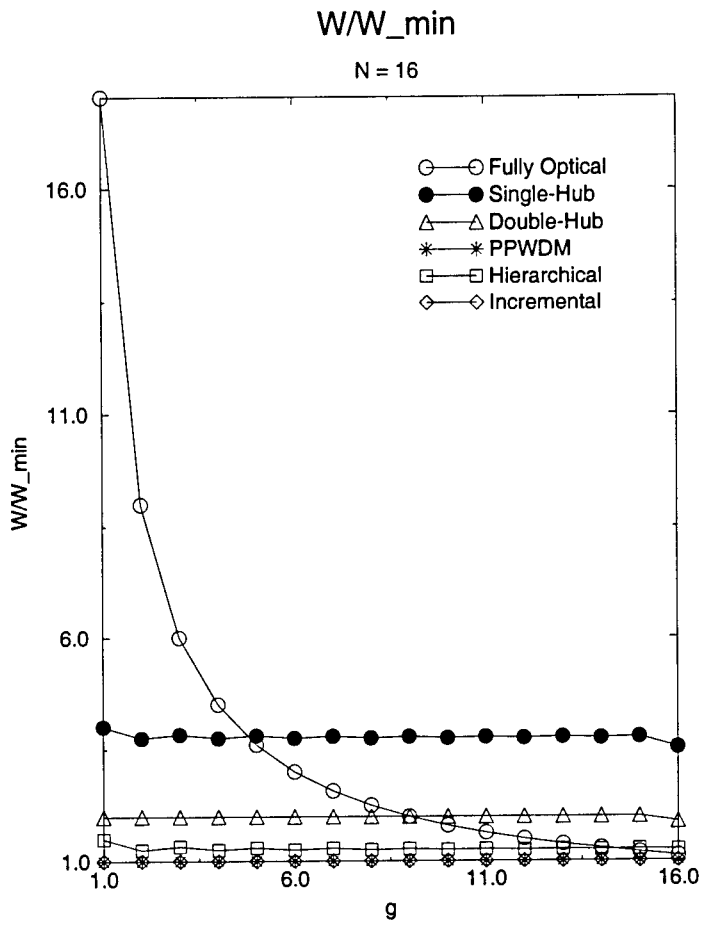



Figure C.14: Relative cost $W/W_{\min}$ for $N = 16$, $c = 16$, and $g = 1, 2, ..., 16$.

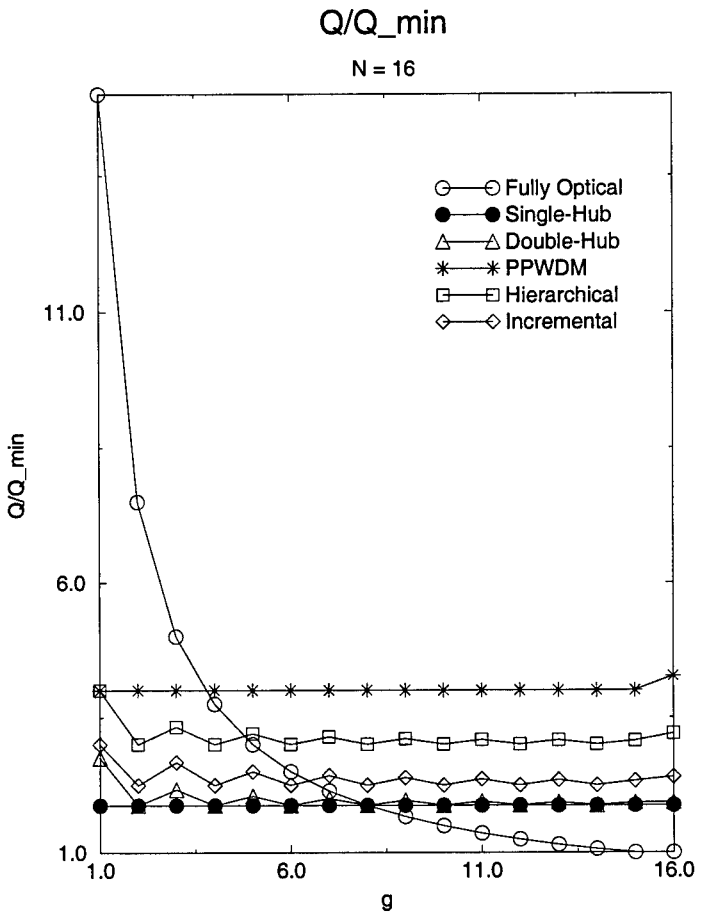



Figure C.15: Relative cost $Q/Q_{\min}$ for $N = 16$, $c = 16$, and $g = 1, 2, ..., 16$.

# Appendix D

# A WDM Ring Network for Incremental Traffic

Galen Sasaki, Ori Gerstel, and Rajiv Ramaswami [1]

**This paper appeared in the Proc. $36^{th}$ Allerton Conference on Communication, Control, and Computing, Monticello IL, September, Sept. 1998.**

**Abstract:** An optical *wavelength division multiplexed* (WDM) ring network is presented that supports dynamic (time-varying) traffic. The traffic streams are at lower rates (e.g., OC-3, 155Mb/s) than the lightpath capacities (e.g., OC-48, 2.5 Gb/s). The network has a nonblocking property if the streams are assumed to never terminate, i.e., *incremental traffic.*

A design algorithm is presented that minimizes the cost of the network, where cost is the number of *add-drop multiplexers* (ADMs). Simulation results are presented comparing the cost of the network with simple lower bounds which show the network has reasonably low cost.

## D.1 Introduction

Optical *wavelength-division-multiplexed* (WDM) networks are expected to become an integral part of telecommunication backbone networks. The networks have fiber optic links, each carrying many communication channels at different wavelengths. The optical

[1]This work was done in part while being with the IBM T.J. Watson Research Center. It was sponsored in part by grant MDA-972-95-C-0001 by DARPA, and was sponsored in part by grant F30602-97-1-0342 by DARPA and Rome Laboratory, Air Force Materiel Command, USAF. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes nothwithstanding any copyright annotation thereon. The views and conclusions contained therein are those of the authors and should not be expressed or implied, of the Defense Advanced Research Projects Agency (DARPA), Rome Laboratory, or the U.S. Government.

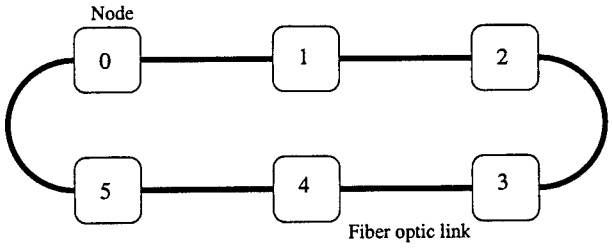



Figure D.1: An optical WDM ring.

channels have high bit rates, e.g., OC-48 (= 2.5 Gb/s) or OC-192 (= 10 Gb/s). At least in the near future, much of telecommunication backbone traffic will be circuit-switched streams being supported by SONET/SDH. These streams will be at a lower-speed (e.g., OC-3 = 155 Mb/s) and must be multiplexed/demultiplexed (i.e., *traffic groomed*) onto and off-of the higher-speed optical channels. It is critical that the network be designed to support expected traffic while keeping network costs at a minimum. In this paper, we will present a WDM ring network architecture that supports *incremental* lower-speed traffic streams, which means that the streams can arrive at arbitrary times but never terminate. (This models the situation when traffic streams are expected to have long holding times, as is usually the case with provisioning high-speed connections today.) The network architecture can be configured to have reasonably low cost.

The network is a WDM ring network (see Figure D.1). The ring topology is viable because SONET/SDH self-healing architectures are ring oriented. The links are bidirectional, each composed of two fibers going in opposite directions. The number of nodes is denoted by $N$ and the nodes are numbered $0, 1, ..., N-1$ in the clockwise direction. The number of wavelengths is denoted by $W$, and the set of wavelengths is denoted by $\{\omega_0, \omega_1, ..., \omega_{W-1}\}$. On each link, each wavelength $\omega_i$ carries a bidirectional (or full-duplex) WDM channel. Notice that a full-duplex WDM channel at a wavelength, say $\omega_i$, on a link is composed of two unidirectional channels at $\omega_i$ being carried by the pair of fibers of the link.

A network node is shown in Figure D.2. It is composed of *add/drop multiplexers* (ADMs) and a single *digital crossconnect system* DCS. The DCS will be assumed to be *wide sense nonblocking*, which means that a traffic stream may be set-up or terminated through it without disturbing existing traffic streams, assuming availability of access ports. The main contribution of cost at a node will be due to the ADMs and DCS.

Notice that optical signals of a WDM channel may be either terminated by an ADM or *passed through* the node to the next link. If terminated by an ADM, the signal is translated into a high-speed electronic bit stream composed of its lower-speed (tributary)

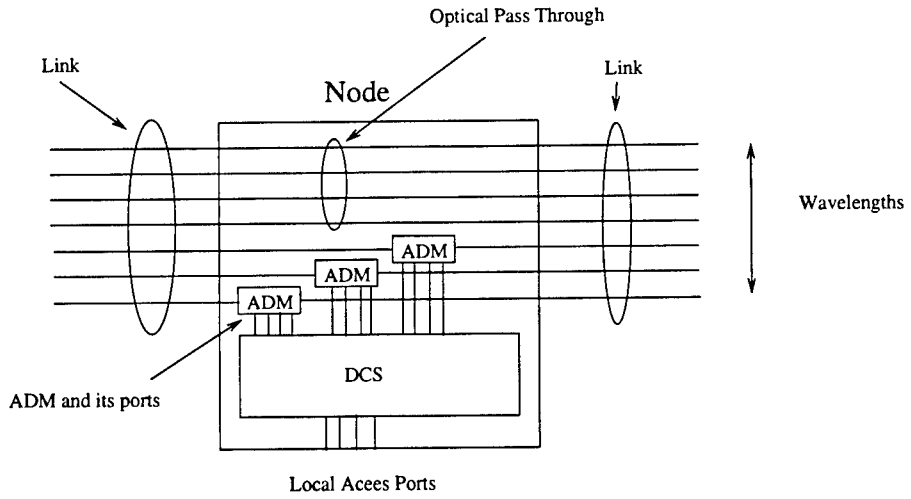



Figure D.2: An optical node.

traffic streams. The lower-speed streams may be added and dropped at the node via the ADM interface ports. The interface ports are connected to the main DCS of the node, where the streams may be crossconnected to other ADMs or local access ports. However, if the optical signals of a WDM channel are passed-through the node, then they continue on the next link at the same wavelength (i.e., no *wavelength conversion*). Notice that the traffic carried by the passed-through optical signals are not intended for the node. Also notice that the network supports optical end-to-end communication connections that are terminated by ADMs and are passed-through intermediate nodes. These full-duplex optical connections will be referred to as *lightpaths*. All lightpaths will have a fixed high-speed bit rate (e.g., OC-48 or OC-192).

The network supports lower-speed traffic streams that are bidirectional and, for simplicity, have the same rate, e.g., OC-3. The number of traffic streams that can be supported by a lightpath is assumed to be some integer denoted by $c$. For example, if the traffic streams are OC-3 (= 155 Mb/s) and the lightpaths are OC-48 (= 2.5 Gb/s) then $c = 16$. The traffic streams are dynamic, so they arrive and terminate at arbitrary times. It is assumed that each traffic stream comes with its own route, i.e., it is *pre-routed*. The network determines which WDM lightpaths the stream uses along its route. Thus, the network has a *lightpath assignment algorithm* (LAA). If a lightpath assignment cannot be found for an arriving traffic stream then it is blocked. The routes for the traffic streams are also assumed to be *simple*, i.e., they traverse any link at most once.

The traffic will be parameterized by an integer $L$ and an integer vector $t = (t(i) : i = 0, 1, ..., N-1)$. At any time, the number of traffic streams over any link is at most $c \cdot L$, assuming no blocking. (In other words, if a traffic stream arrives then the number

of other traffic streams using any link along its route is less than $c \cdot L$.) This will be referred to as the *Link Load L Constraint.* In addition, each node $i$ may terminate at most $c \cdot t(i)$ traffic streams from the clockwise or counter-clockwise direction along the ring. Thus, node $i$ can terminate up to $2c \cdot t(i)$ traffic streams but then half must come from the clockwise direction and the other half must come from the counter-clockwise direction. This will be referred to as the *Nodal Load t Constraint.* Notice that to insure no blocking we should have $L \leq W$ and for each node $i$, there should be at least $t(i)$ ADMs.

In Section D.2 we present our network, we refer to as the *Incremental Ring Network.* It is wide-sense non-blocking for incremental traffic. Thus, an arriving traffic stream will be able to find a lightpath assignment without disturbing existing traffic streams (under the Link Load and Nodal Load Constraints). We also present a design algorithm that configures the network to minimize cost. The configuration involves placing ADMs at nodes at different wavelengths which, in turn, determines the lightpaths of the network.

The network cost will be the sum of costs at each node. We will assume that the cost of each node $i$ is dominated by the ADM and DCS costs. We will assume that the ADM cost is proportional to the number of ADMs at node $i$, which we denote by $A(i)$. We will assume that the DCS cost is a function of $A(i) + t(i)$ because $c(A(i) + t(i))$ is an upper bound on the number of (lower-speed) ports at the DCS. For example, the DCS cost could be a quadratic function of $A(i) + t(i)$. Thus, we will assume that the cost at node $i$ is a function $D(A(i), t(i))$, and therefore the cost of the network is $\sum_i D(A(i), t(i))$.

Simulation results are presented in Section D.3. The results show the performance of the design algorithm for randomly generated values of $t$. In Section D.4 we state our conclusions.

For the remainder of this section, we will discuss related work. First, we should mention that this paper is a continuation of [15], where a number of WDM ring networks were considered and electronic (ADM) costs were compared. The Incremental Ring network was introduced briefly, but without any analysis or design algorithm. In addition, it was constrained to a uniform traffic model ( $t(i)$ is the same for all nodes $i$).

ADM costs were also considered in [14, 24, 30], but where traffic was assumed to be static. In [14], the traffic was a collection of lightpaths to be set up in a WDM ring network. In [24, 30], the traffic was a *uniform* pattern of lower-speed traffic streams. The static traffic pattern of all traffic streams terminating at one *central* node was also considered in [24].

We should finally note that typically researchers have concentrated on numbers of wavelengths, congestion, delay, or probability of blocking. There is previous work on

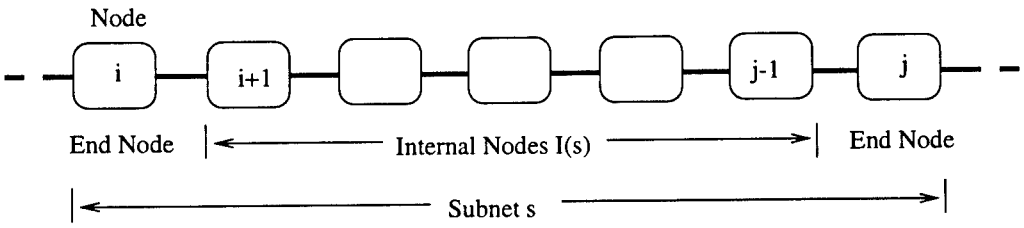



Figure D.3: The segment of subnet $s$.

WDM network design for lower-speed traffic streams [2, 7, 12, 27, 28], assuming traffic is *static*. There are also a number of papers on WDM networks with dynamic traffic (e.g., [4, 1, 17, 20, 26]), but where lightpaths are switched and lower speed traffic streams are not considered.

## D.2 Incremental Ring

In this section we present the *Incremental Ring network*, and will assume incremental traffic. The network has $W = L$ wavelengths which is the minimum to insure no blocking under the Link Load $L$ Constraint. We will first describe the network architecture, and then show that it is nonblocking for incremental traffic. At the end of the section, we will present a strategy to configure the architecture to minimize cost.

The Incremental Ring network is organized with respect to pieces of it called *subnets*. A subnet $s$ is composed of a sequence of links (and nodes) along the ring in the clockwise direction. This sequence will be referred to as its *segment*. The subnet's *length* is the length of its segment. The subnet's *end nodes* are the nodes at the end of its segment. The rest of its nodes are referred to as the *internal nodes* and denoted by $I(s)$. Figure D.3 shows a segment and its end and internal nodes.

A subnet $s$ is also composed of a collection of wavelengths and lightpaths. The wavelengths are $w_0, w_1, ...., w_{r(s)-1}$, where $r(s)$ denotes the number of wavelengths for $s$. The lightpaths only use the wavelengths and links of $s$.

The Incremental Ring architecture is organized as a *tree* of subnets. One of the subsets is referred to as the *root* and its segment includes all the links of the ring. The root's segment starts and ends at some node, which we refer to as the *root node*. The number of wavelengths for the root is $W$, i.e, $r(root) = W$.

If a subnet $s$ has length greater than one (i.e., it has at least one internal node) then it is referred to as a *parent* subnet. Note that the *root* subnet is a parent if $N > 1$. A parent subnet $s$ has two *children* subnets, say $s_0$ and $s_1$. The segments of the children are defined by bisecting the segment of $s$ at some internal node denoted by $b(s) \in I(s)$.

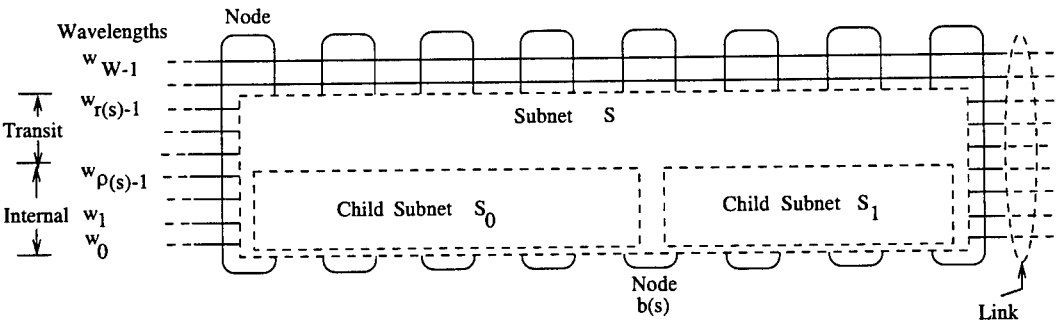



Figure D.4: A parent subnet $s$ composed of its two children subnets $s_0$ and $s_1$.

The resulting subsegments are the segments for $s_0$ and $s_1$. The number of wavelengths assigned to each of the children is $r(s_0) = r(s_1) = \rho(s)$, where $\rho(s)$ is defined to be

$$\rho(s) = \min\{r(s), \sum_{i \in I(s)} t_B(i)\}. \tag{D.1}$$

Note that $\rho(s)$ is equal to

$$\min\{W, \sum_{i \in I(s)} t_B(i)\} \tag{D.2}$$

since $r(root) = W$ and for an arbitrary parent subnet $s'$ and its child $s^*$, $\sum_{i \in I(s')} t_B(i) \geq \sum_{i \in I(s^*)} t_B(i)$. Figure D.4 shows an example subnet $s$, its bisecting node $b(s)$, and two children $s_0$ and $s_1$. As shown in the figure, the $r(s)$ wavelengths of $s$ are categorized into two types for $s$: *internal* and *transit*. The internal wavelengths of $s$ are $\{w_0, w_1, ..., w_{\rho(s)-1}\}$, and the transit wavelengths of $s$ are $\{w_{\rho(s)}, w_{\rho(s)+1}, ..., w_{r(s)-1}\}$. Thus, the $\rho(s)$ internal wavelengths of $s$ are the wavelengths of its children. This categorization of wavelengths will be used later to define where lightpaths are placed in the subnet (or equivalently where ADMs are placed), and how lightpaths are assigned to traffic streams.

If a subnet $s$ has only one link then it has no children and is called a *leaf* subnet. Leaf subnets only have transit wavelengths. Now note that the collection of leaf and parent subnets form a tree of subnets, where the root and leaf subnets are the root and leaves, respectively, of the tree. Also note that a subnet can be both a child (of some subnet) and a parent (of two other subnets).

Next, we define how ADMs are placed at each node, which will in turn define the placement of lightpaths. First, the root node has $W$ ADMs terminating each wavelength on both of its links. For all other nodes, the placement is as follows. A node must be a bisecting node $b(s)$ for some parent subnet $s$. The node $b(s)$ has $\rho(s)$ ADMs, which terminate wavelengths $w_0, ...w_{\rho(s)-1}$ on each of the node's links. Thus, for subnet $s$, its end and bisecting nodes terminate lightpaths on its internal wavelengths, while *only* its

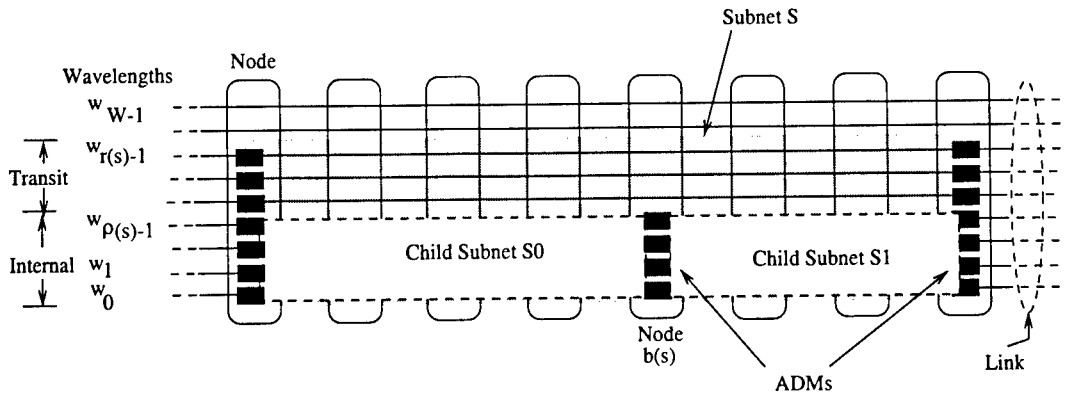



Figure D.5: A parent subnet $s$ composed of its two children subnets $s_0$ and $s_1$ and the placement of ADMs.

end nodes terminate lightpaths on its transit wavelengths (refer to Figure D.5). Thus, lightpaths on its transit wavelengths directly connect its end nodes, and cannot be used for traffic streams that terminate at its internal nodes.

To complete the description of the Incremental Ring architecture, we define its *lightpath assignment algorithm* (LAA). The LAA takes an incoming traffic stream with a route, and finds a contiguous sequence of lightpaths along the route that can carry the stream. To define the LAA for the ring, we will define LAAs for the subnets.

The LAA for a subnet takes a route and assigns a contiguous sequence of its lightpaths along the route that have spare capacity. Thus, a traffic stream following the route can be carried by the lightpaths. The route is assumed to be entirely in the segment of the subnet. Thus, the route may actually be part of some larger traffic stream route that goes through the subnet. There are two kinds of routes: *internal* and *transit*. An internal route ends at at least one internal node of the subnet, while a transit route does not, i.e., it ends at the end nodes.

We will describe the LAA for a subnet $s$ by first assuming that $s$ is a parent subnet, as shown in Figure D.4. How the LAA assigns lightpaths to a route depends on whether the route is internal or transit. If the route is internal then it is first split at the bisecting node $b(s)$. (Of course, it is not split if it does not pass through $b(s)$.) The resulting subroutes fit into the segments of the children subnets of $s$. The LAAs of the children are used to assign lightpaths to the subroutes. These assignments combine to provide the assignment for the original route. However, if the route is transit, then the LAA will first attempt to assign a lightpath on a transit wavelength of $s$ to the route. If all such lightpaths have no spare capacity then the LAA assigns lightpaths to the route as if it were an internal route, i.e., it splits the route at $b(s)$ and uses the LAAs of the children

subnets.

The LAA for a leaf subnet $s$ is simpler because all routes and wavelengths are transit. A route is assigned to a lightpath of $s$ with spare capacity.

We have completed the definition of the LAAs for subnets. The LAA for the root subnet is basically the LAA for the Incremental Ring. In particular, the LAA for the Incremental Ring will assign lightpaths to an incoming traffic stream as follows. It first splits the route of the stream at the root node (if the route passes through the root). The resulting subroutes are entirely in the root subnet and can be assigned lightpaths by the subnet's LAA. This lightpath assignment is an assignment for the traffic stream since the lightpaths that meet at the root node can be crossconnected.

Next we will prove the following property.

**Theorem 6** *Consider an Incremental Ring network under Traffic Assumption B and incremental traffic. Then the Incremental Ring network is wide sense nonblocking.*

The theorem is implied by the next lemma, which uses the following definitions. Consider a subnet $s$. The subnet is said to have *spare capacity* in a link if one or more of its wavelengths have spare capacity in the link. The subnet is said to be *wide sense nonblocking* if given a route, the LAA of the subnet will find a lightpath assignment for the route as long as the subnet has spare capacity in each link along the route. Note that having the spare capacity does not preclude a valid lightpath assignment because in order for a set of lightpaths to carry a traffic stream they must terminate at the ends of the route, and at common intermediate nodes. Note that the next lemma implies Theorem 6 because it is applicable to the root subnet.

**Lemma 2** *Consider an Incremental Ring network with incremental traffic. Then each subnet $s$ of the ring is wide sense nonblocking.*

*Proof.* To prove the lemma for leaf subnets is trivial since the subnet has only one link. Thus, we will only consider parent subnets. The proof of the lemma is by induction. Consider a parent subnet $s$ and its two children subnets $s_0$ and $s_1$. We will assume that the children subnets are nonblocking, and proceed to show that $s$ is also nonblocking.

Let us see how subnet $s$ assigns lightpaths to traffic streams over time. Initially, there are no traffic streams so all lightpaths in the subnet are empty. As traffic streams arrive they are assigned lightpaths so that internal traffic (i.e., streams that terminate at internal nodes of $s$) are assigned to lightpaths with internal wavelengths, and transit traffic (i.e., streams that do not terminate at internal nodes of $s$) are assigned to lightpaths with transit wavelengths.

Note that this can continue while transit traffic does not completely fill lightpaths with transit wavelengths. This is partly due to the traffic model which assumes that each internal node $i \in I(s)$ can terminate at most $c \cdot t_B(i)$ traffic streams from the clockwise or counter-clockwise direction. This implies that for any link of the subnet, the number of internal traffic streams that go over it is at most $c \cdot \rho(s)$. The internal traffic can be accommodated by the lightpaths with internal wavelengths because

- children subnets are assumed to be nonblocking and occupy the $\rho(s)$ internal wavelengths; and
- bisecting node $b(s)$ (which is between the children subnets) terminates all the internal wavelengths with ADMs, and can therefore crossconnect the spare capacity in the two children subnets.

Thus, while transit streams do not completely fill the transit wavelengths, internal traffic streams will be accommodated by lightpaths at internal wavelengths. Hence, there will be no blocking.

Now at some point the lightpaths with transit wavelengths may become completely filled. Then the only spare capacity left in the subnet $s$ is in its internal wavelengths. Arriving traffic streams will not be blocked from this spare capacity because (i) the spare capacity resides in the nonblocking children subnets, and (ii) the bisecting node $b(s)$ can crossconnect the spare capacity in the two children subnets. □

Finally, we will describe a strategy to place ADMs (and lightpaths) to minimize cost. Notice that the placement of ADMs is dependent on the choice of root and bisecting nodes. Thus, the strategy is to find an optimal choice of these nodes. Before presenting the strategy, we give some definitions.

- Let $\sigma(i,k)$ denote a segment of the ring that starts at node $i$ and goes clockwise over $k$ links.
- Let $I'(i,k)$ denote the *internal* nodes of $\sigma(i,k)$, i.e., $I'(i,k) = \{(i+j) \bmod N : 0 < j < k\}$.
- Consider all possible ways of configuring the Incremental Ring, i.e., all possible ways of choosing the root and bisecting nodes. Of these, consider all configurations that have a subnet with segment $\sigma(i,k)$. Let $\mathcal{C}(i,k)$ denote these configurations of the Incremental Ring.
- For each configuration $\gamma \in \mathcal{C}(i,j)$, let $q_\gamma(i,k)$ be the sum of costs over the internal nodes $I'(i,k)$ assuming the Incremental Ring is configured as $\gamma$. Let $q(i,k) =$

$\min_{\gamma \in \mathcal{C}(i,j)} q_\gamma(i,k)$. Thus, $q(i,k)$ is the minimum sum of costs at the internal nodes $I'(i,k)$ assuming that $\sigma(i,k)$ is the segment of a subnet.

Notice that for each node $i$, $D(W,t(i)) + q(i,N)$ is the minimum cost for the Incremental Ring assuming $i$ is the root node because the root has cost $D(W,t(i))$ and the minimum cost of the other nodes is $q(i,N)$. Thus, the minimum cost for the Incremental Ring (over all possible roots) is $\min_i[D(W,t(i)) + q(i,N)]$.

We now describe the strategy to determine the root and bisecting nodes. It consists of two steps. The first step is to compute the values for $q(i,k)$ for all nodes $i$ and all $k = 1, 2, ..., N$ by using the equation

$$q(i,k) = \begin{cases} 0, & \text{if } k = 1 \\ \min_{i<j<k} \{q(i,j) + q((i+j) \bmod N, k-j) + d(i,j)\} & \text{if } k > 1 \end{cases} \tag{D.3}$$

where

$$d(i,j) = D\left(\min\{W, \sum_{n \in I'(i,k)} t_B(n)\}, \quad t((i+j) \bmod N)\right) \tag{D.4}$$

The Equation (D.3) is true for $k = 1$ because $\sigma(i,1)$ corresponds to a leaf subnet, which has no internal nodes (and no ADMs or DCS). The equation is also true for $k > 1$ because (a) the minimum is over all bisecting nodes $(i+j) \bmod N$ for a subnet with segment $\sigma(i,k)$, (b) the terms $q(i,j)$ and $q(i+j, k-j)$ are the minimum cost at internal nodes of the two children of the subnet, and (c) the term

$$D\left(\min\{W, \sum_{n \in I'(i,k)} t_B(n)\}, t((i+j) \bmod N)\right)$$

is the cost at the bisecting node where $\min\{W, \sum_{n \in I'(i,k)} t_B(n)\}$ is the number of its ADMs. The values for $\{q(\cdot,\cdot)\}$ are determined by first computing $q(i,2)$ for all nodes $i$. Then computing $q(i,3)$ for all nodes $i$, and so forth.

The second step is to determine the root and bisecting nodes that minimize the cost of the network from the values of $\{q(\cdot,\cdot)\}$. This is straightforward. In particular, the node $i$ that minimizes $\min_i q(i,N)$ is the optimal root node. The node $(i+j) \bmod N$ that minimizes the right hand side of (D.3) is the optimal bisecting node for the corresponding subnet.

## D.3 Simulation Results

In the previous section, a design algorithm was given to determine the placement of ADMs for the Incremental Ring Network. In this subsection, simulation results of the

algorithm will be given. For simplicity, the cost of a node will be the number of its ADMs. Thus, DCS costs are ignored. We will compare the cost resulting from the design algorithm with the simple lower bound $\frac{1}{N}\sum_{i=0}^{N-1} t(i)$. For the simulations, we assume that $W = 32$ and $c = 16$ (e.g., 32 OC-48 WDM channels per link, and traffic streams are OC-3).

Each simulation run corresponds to a randomly generated set of values for $(t(i) : i = 0, 1, ..., N-1)$. This is determined by a single parameter $F \geq 1$. To generate the values, first a collection $f = (f_i : i = 0, 1, ..., N-1)$ of real values are randomly generated. The values are independent and uniformly distributed over the interval $[1, F]$. The value $f_i$ represents the relative frequency that node $i$ will terminate a traffic stream. For example, if $f_i/f_j = 2$ then node $i$ will be twice as likely to terminate a traffic stream as node $j$. Next, given $f$, a collection of traffic streams is generated. The collection will fill each link with exactly $c \cdot W$ streams, thereby maximally loading each link. The streams are generated sequentially and go in the clockwise direction around the ring. The first stream starts at a random node. It is routed clockwise and its end node will be at most $N/2$ hops away (to model shortest path routing of streams). Its end node is chosen randomly according to $f$. Thus, node $i$ will be $f_i/f_j$ more likely to be chosen than node $j$. There is the exception that the node that is exactly $N/2$ hops away has its frequency decreased by a half, to take into account that a shortest hop path between two nodes at opposite ends of the ring could be on either side of the ring. Now the end node of the first traffic stream becomes the starting node of the second traffic stream. The end node of the second traffic stream is randomly chosen using $f$. The end node of the second traffic stream becomes the starting node of the third traffic stream, and so on. This continues until all bandwidth in each link is accounted for. After the collection of traffic streams has been generated, each $t(i)$ is computed. In particular, if $g(i)$ is the number of traffic streams terminating at node $i$ from the clockwise (or counter clockwise) direction on the ring, then $t(i) = \lceil g(i)/c \rceil$.

We ran simulations for $N = 8, 12$, and 16 for different values of $F$. Figure D.6 shows the average values of

- $ADMAVG$ = the average number of ADMs per node,
- $tAVG$ = the average value of $t(i)$ (i.e., $\frac{1}{N}\sum_{i=0}^{N-1} t(i)$), which is a lower bound on the average number of ADMs per node.
- $tMIN$ = the minimum value of $t(i)$ (i.e., $\min_i t(i)$),
- $tMAX$ = the maximum value of $t(i)$ (i.e., $\max_i t(i)$).

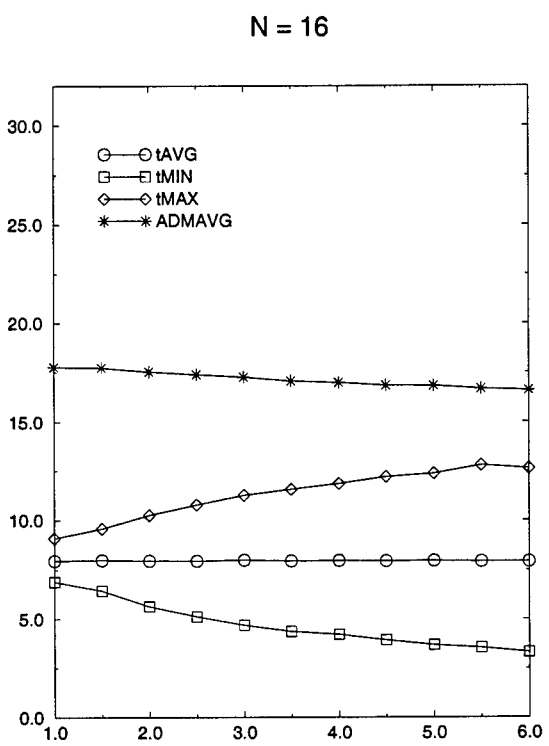



Figure D.6: For a ring network with $N = 16$ and $W = 32$, the plots give the average values of $tAVG$, $tMIN$, $tMAX$, and $ADMAVG$.

as functions of $F$ for $N = 16$, where each data point is an average over 100 random instances.

Note that as $F$ increases the difference in value between $tMIN$ and $tMAX$ increases, as expected. In addition, the number of ADMs required decreases slightly.

Figures D.7 and D.8 show the ratios $\frac{ADMAVG}{tAVG}$ and $\frac{ADMAVG}{W}$, respectively, for $N = 8, 12$, and 16. The first ratio indicates the cost (in ADMs) of the Incremental Ring relative to the simple lower bound. Note that it is approximately two, and thus traffic streams will on average traverse approximately two lightpaths. This is reasonably low especially if traffic streams must be groomed to utilize bandwidth. The other ratio $\frac{ADMAVG}{W}$ indicates a cost savings due to optical signals being allowed to pass through nodes (i.e., not terminating every wavelength). Note that the cost savings is at least 27%, 36%, and 44% for $N = 8, 12$, and 16, respectively (see Figure D.8). This can be considerable (e.g., if ADMs cost \$100,000 per node then a 27% savings amounts to saving \$864,000 per node, and a 44% savings amounts to saving \$1.4 million per node). Both Figures D.7 and D.8 show that the cost of the Incremental Ring improves as $F$ increases.

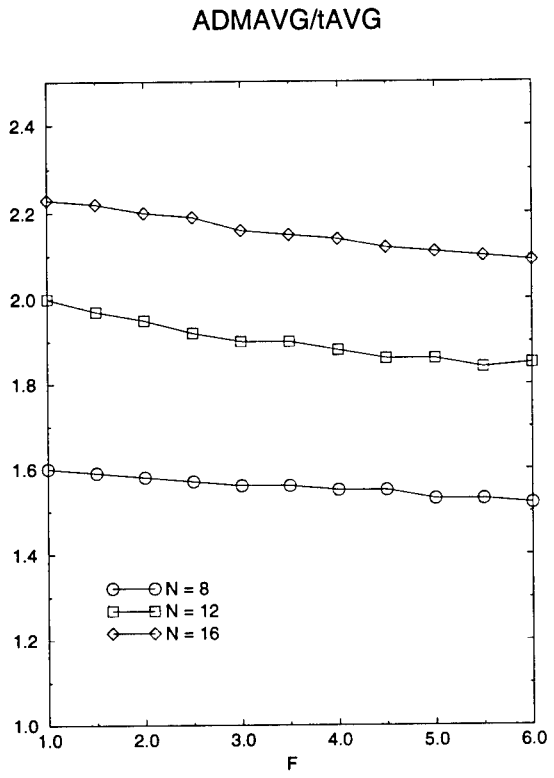



Figure D.7: The ratio $ADMAVG/tAVG$ as a function of $F$ for three different values of $N$. The curves from top to bottom are for $N = 16, 12$, and 8, respectively.

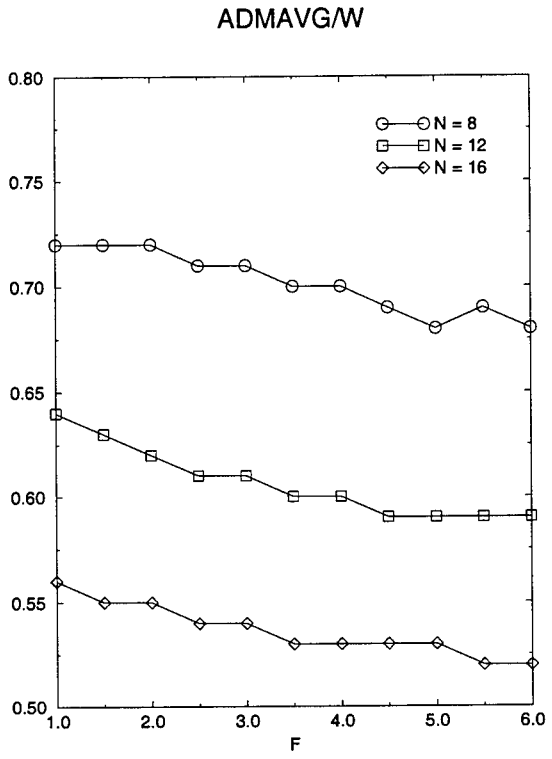



Figure D.8: The ratio $ADMAVG/W$ as a function of $F$ for three different values of $N$. The curves from top to bottom are for $N = 8, 12$ and 16, respectively.

## D.4 Conclusions

We described the Incremental Ring in detail and presented its wide sense nonblocking property for incremental traffic. Its ADM cost is reasonably low, approximately twice the value of a very simple lower bound, and shows considerable savings over networks that do not allow optical signals to pass through nodes.

# Appendix E

# A Minimal Cost WDM Network for Incremental Traffic

Galen Sasaki and Tao Lin

**This paper appeared in the Proc. SPIE Conference on All-Optical Networking 1999: Architecture, Control, and Management Issues, Sept. 1999. It was a co-winner of the BEST PAPER award.**

**Abstract.** A *wavelength division multiplexed* (WDM) network is considered for arbitrary topologies. The network allows optical signals to pass through nodes, which often results in less electronic and opto-electronic equipment than networks that do not, i.e., networks that only switch traffic electronically. A disadvantage of having optical pass through is that there is less capability to switch tributary traffic streams and so there may be more blocking. However, it is shown that the network operates as well as a network with only electronic switching under a particular *incremental* traffic model. Examples are given that shows that the network can have lower cost when the cost is dominated by the *line terminating equipment*. A simple heuristic design algorithm is also given to configure the network to minimize its cost.

## E.1 INTRODUCTION

*Wavelength division multiplexing* (WDM) is the key technology platform for backbone networks. It is a *frequency division multiplexing* (FDM) approach to exploit the available bandwidth on optical fibers. Each channel is carried on a *wavelength*, which is equivalent to a carrier frequency. These channels carry data at very high rates (e.g., OC-48 = 2.5 Gbps, or OC-192 = 10 Gbps). In many cases, these optical channels carry bit streams using the SONET/SDH format. Thus, the traffic that is being supported is circuit switched (or constant bit rate) traffic.

Currently, WDM technology is being deployed in links. It provides a way to make a single optical fiber link operate as many multiples of *virtual fibers*, each carrying a different optical signal at different wavelengths. Figure E.1 shows a node with three incident fiber links and four wavelengths. Each of the WDM channels are terminated by *line terminating equipment* (LTE). LTEs convert optical signals to electronic bit streams and vice versa. They also multiplex and demultiplex lower-speed traffic streams onto the WDM channels. For example, the traffic streams supported by the network may be OC-3 (= 155 Mbps). If the WDM channels are at the OC-48 rate then the LTE can multiplex up to 16 traffic streams onto a channel.

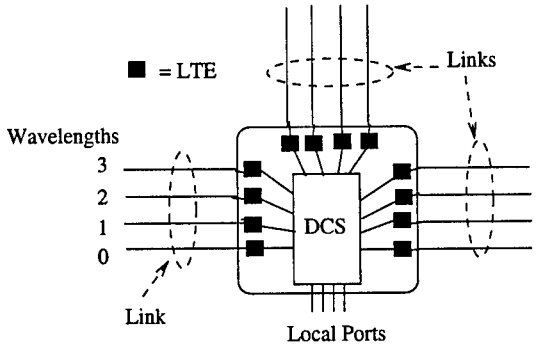



Figure E.1: An opaque node.

If, at a node, each of the WDM channels is terminated by an LTE then the traffic streams must be cross connected by a switch, referred to as a *digital crossconnect system* (DCS), as shown in Figure E.1. This type of node will be referred to as *opaque* because the optical signals do not pass through the node. The advantage of an opaque node is that the technologies involved are proven, and the node has the maximum of switching resources to efficiently use the bandwidth.

Another network node is shown in Figure E.2. It also has three incident fiber links, and four wavelengths. Unlike the opaque node, it allows optical signals to pass without changing their wavelengths. We refer to this type of node as an *optical* node. Note that the channels at wavelengths $\{0, 1\}$ are terminated by LTEs. The channels at wavelengths $\{2, 3\}$ have their signals go through *optical crossconnects* (OXCs), which switch the optical signals without converting them to electronic signals. The optical signals at these wavelengths may pass through an OXC from link to link, or they may be switched through the OXC to an LTE. All LTEs (and local ports) are connected to a single DCS to switch traffic streams between them. The advantage of an optical node is that there is less electronic and opto-electronic equipment (LTEs and DCSs), which can dominate cost.

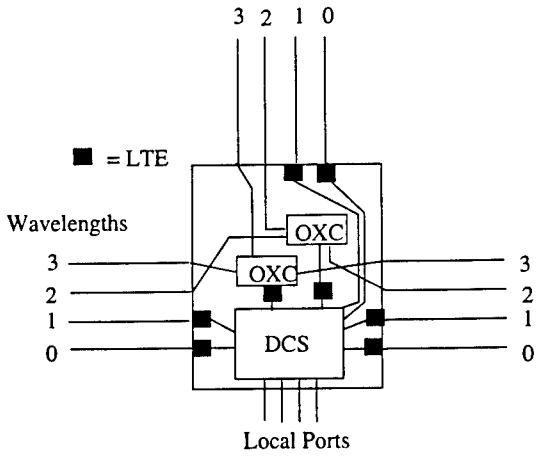



Figure E.2: An optical node.

In this paper, we will consider a particular network with a mix of optical and opaque nodes which we will refer to as the *optical* network. It will be compared to another network, where all nodes are opaque. It will be shown that the optical network can support the same traffic as the opaque network given extra wavelengths and certain restrictions on the traffic dynamics. The number of extra wavelengths is an increasing function of the maximum number of links incident to any node. One of the traffic restrictions is that traffic streams can arrive at any time, but can never terminate. In this sense, the traffic is *incremental.* This assumption models the fact that high speed connections (e.g., OC-3 = 155 Mbps) are rarely terminated because they support large numbers of users.

Section E.2 describes the network and traffic models. It also has simulation data of the cost of an optical network with 14 nodes, where cost is the number of LTEs. The cost of the optical network is compared with the cost of a network with only opaque nodes. The percentage reduction in cost can be as high as 17%. Section E.3 shows that the optical network is nonblocking under the traffic model. In Section E.4, a design problem is investigated which is to configure the network to minimize cost. The problem is inherently difficult, and so a heuristic is presented and evaluated. A brief summary is given in Section E.5. We should finally note that the results of this paper are an extension of our earlier work [15, 29] which focused on ring topology networks.

## E.2 NETWORK AND TRAFFIC MODELS

The optical network has a set of nodes and a set of bidirectional fiber links. Each fiber link has $W$ wavelengths numbered $0, 1, ..., W-1$, where $W$ is some integer. The network

supports *constant bit rate* (CBR) traffic streams all at the same rate (e.g., OC-3). An integer parameter $g$ (for granularity) denotes the number of traffic streams that can be multiplexed onto a WDM channel. For example, if the WDM channels are at OC-48 rates and the traffic streams were OC-3 then $g = 16$.

Bidirectional traffic streams are offered to the network at arbitrary times, but they never terminate. There is an integer parameter $L$ for the traffic which is the maximum traffic load offered on any link. In particular, the route of any offered traffic stream must be along links where the number of existing streams is less than $g \cdot L$. We call this the *link traffic constraint.* This means that if $W \geq L$ then each of those links must have some channel that is not completely full. (Note that in an optical network, this does not preclude that the traffic stream will be accepted because there may be insufficient LTEs to switch the stream between wavelengths.)

For each node $i$, there is an integer parameter $t(i)$ that restricts the amount of traffic streams that terminate at the node. In particular, node $i$ can terminate at most $g \cdot t(i)$ traffic streams through any of its incident links. Thus, the total number of traffic streams that can terminate at node $i$ is at most $d(i) \cdot g \cdot t(i)$, where $d(i)$ is the number of links incident to the node, i.e., the node's *degree.* (For the case of ring networks, $d(i) = 2$ for all nodes $i$.) This constraint is called the *node traffic constraint*, and models traffic where a significant fraction of the traffic bypasses nodes.

The network has two types of nodes: *opaque* and *optical.* We will describe these nodes in more detail a little later. In the network, each optical node has only opaque neighbors as shown in Figure E.3. Therefore, the set of optical nodes form an *independent set* in graph theory, i.e., a subset of nodes that are not neighbors of each other. The number of incident fiber links on the optical nodes determines the value of $W$. In particular, $W$ equals $L + d^* - 2$, where $d^* = \max_{i \in O} d(i)$ and $O$ is the set of optical nodes.

Each fiber link has $W$ wavelengths, but not all of them are necessarily *used.* If the wavelength is *used* then its channel carries optical signals; otherwise, it is *unused.* The wavelengths used on a link depend on its end nodes. If it has two opaque nodes then $L$ of its wavelengths are used which are from the set $\{0, 1, ..., L-1\}$. If it has an optical end node $i$ then it has $L+d(i)-2$ used wavelengths which are from the set $\{0, 1, ..., L+d(i)-3\}$.

Now we describe the internal architecture of opaque and optical nodes. The opaque nodes terminate WDM channels at used wavelengths with LTEs, as shown in Figure E.1. The LTEs are connected to each other and local ports by a DCS.

An optical node $i$ will have its $L - d(i) - 2$ used wavelengths partitioned into two types, referred to as *transit* and *local.* Transit wavelengths carry traffic that bypass the node, while local wavelengths can carry traffic that either bypass or terminate at

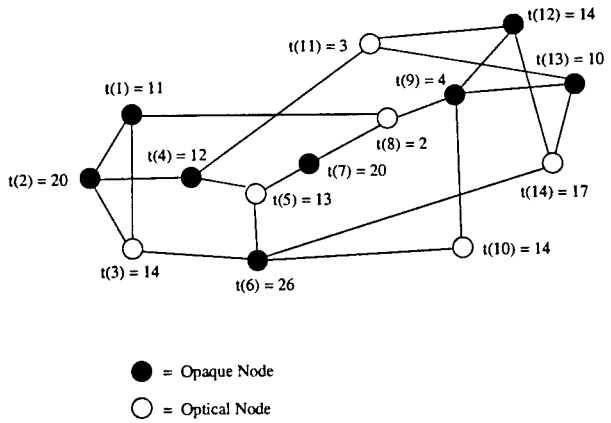



Figure E.3: NSFNET.

the node. The $t(i)$ wavelengths $\{0, 1, ..., t(i) - 1\}$ are the local wavelengths, and the other $L - d(i) - 2 - t(i)$ wavelengths are the transit wavelengths. The channels at the local wavelengths are all terminated by LTEs. The channels at the transit wavelengths are connected through OXCs. In particular, the channels at a transit wavelength are connected through an OXC dedicated to the wavelength. The OXC is also connected to $d(i) - 2$ LTEs. Thus, the total number of optical crossconnects is $L + d(i) - 2 - t(i)$, the total number of LTEs for local wavelengths is $d(i)t(i)$, and the total number of LTEs for transit wavelengths is $(d(i) - 2)(L + d(i) - 2 - t(i))$. The LTEs and local ports are all connected to a DCS. Figure E.2 shows an optical node $i$ with three incident links, four wavelengths (all of which are used), and $t(i) = 2$.

We will evaluate the cost of the network assuming that the LTEs dominate. Thus, the cost is the number of LTEs. For an opaque node $i$, the number of LTEs is $L \cdot d(i) + \sum_{k \in M(i)}(d(k) - 2)$, where $M(i)$ is the set of optical neighbors of $i$. For an optical node $i$, the number of LTEs is $t(i)d(i) + (L + d(i) - 2 - t(i))(d(i) - 2)$. The total number of LTEs can be written as

$$\sum_{i \in V} L \cdot d(i) + \sum_{i \in O}[-2(L - t(i)) + (d(i) - 2)(2d(i) - 2)], \tag{E.1}$$

where $V$ is the set of nodes, and $O$ is the subset of optical nodes. Notice that the LTE cost can be minimized by finding a set of optical nodes $O$ that minimizes the second sum. The minimization problem is basically to find a maximum weighted independent set.

To conclude the section, we present simulation results of the cost of the network, where the cost is the number of LTEs. Figure E.4 shows the simulation results for the NSFNET topology, as shown in Figure E.3, and when $L = 32$. Notice that since the topology has at most four incident links on any node, $W \leq L + 2$.

| Range of $t(i)$ | Min Cost | Reduced Cost | Percentage Reduction |
|---|---|---|---|
| [16..16] | 1168 | 176 | 13.1% |
| [14..18] | 1164 | 180 | 13.4% |
| [12..20] | 1152 | 192 | 14.3% |
| [10..22] | 1144 | 200 | 14.9% |
| [8..24] | 1140 | 204 | 15.2% |
| [6..26] | 1130 | 214 | 15.9% |
| [4..28] | 1120 | 224 | 16.7% |
| [2..30] | 1112 | 232 | 17.3% |

Figure E.4: Table showing the average minimum cost for different ranges of values of $\{t(i)\}$.

The values of $\{t(i)\}$ were randomly and uniformly chosen within a range. For each set of $\{t(i)\}$, a minimum cost was computed by exhaustively examining all possible sets of optical nodes. The table in the figure has four columns. The first column gives the range of values of $\{t(i)\}$. Note that the average value of $\{t(i)\}$ is always 16, which is half of $L$. The second column "Min Cost" gives the average minimum cost over 100 random samples. The third column "Reduced Cost" gives the average reduction in cost compared to a network with only opaque nodes. Note that a network with only opaque nodes will require $W = L$ wavelengths and 1344 LTEs to insure no blocking of offered traffic streams, under the link and node traffic constraints. The reduced cost is the minimum cost of optical network (of a mix of opaque and optical nodes) minus 1344. The fourth column "Percentage Reduction" is the percentage of reduction, which is equal to $\frac{Reduced\ Cost}{Min\ Cost} \times 100\%$. Figure E.4 shows that the more variability in $\{t(i)\}$ the greater the reduction in cost. Figure E.3 shows an example solution when $\{t(i)\}$ is chosen within the range [2..30]. For this solution, the number of wavelengths required is $L + 1$, which is just one more than what is needed for a network with only opaque nodes to insure no blocking of offered traffic.

## E.3 NONBLOCKING RESULTS

We will proceed to prove the following theorem.

**Theorem 1.** *There is an algorithm to set up the traffic streams in the network to avoid blocking for incremental traffic assuming the link and node traffic constraints.*

We will describe the algorithm to set up a traffic stream. First, the new stream is

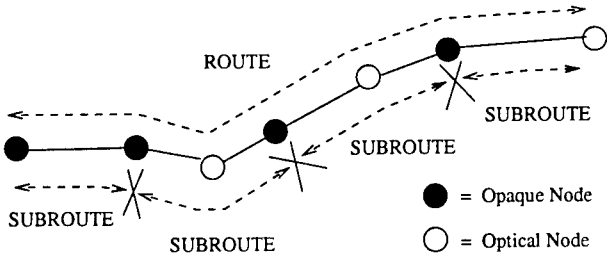



Figure E.5: A traffic stream route.

routed using an algorithm such that the number of existing streams on any link along the route is less than $g \cdot L$, i.e., the new stream satisfies the link traffic constraint. To assign the stream to WDM channels along the route, the route is first split into subroutes, where the splitting is done at opaque nodes as shown in Figure E.5. The subroutes are assigned to channels by an algorithm, discussed below. Then these assignments are pieced together to form the assignment for the entire stream. Note that the composed assignment is valid for the stream because at any opaque node (where splitting takes place) traffic from all its incident WDM channels are switched through a common DCS.

Next we describe how to assign a subroute to WDM channels. There are two types of subroutes. The first kind traverse a single link with two opaque end nodes. These are assigned to any available channel on the link. Note that a stream will never be blocked at these links due to the link traffic constraint.

The second kind of subroute has either one or two hops and has an optical node. If it has one hop then an optical node is at one of its ends, and if it has two hops then an optical node is in its middle. Note that such a subroute is completely on some subnetwork, which has a single optical node and the node's opaque neighbors. Such a subnetwork forms a *star* topology as shown in Figure E.6 with an optical node *hub* connected to opaque neighbors. So we can focus our attention on such a star subnetwork.

Consider a star topology subnetwork with an optical node $i$ as its hub. Recall that there are $t(i)$ local wavelengths, and these are $\{0, 1, ..., t(i) - 1\}$. These wavelengths are terminated by LTEs at node $i$ and the opaque nodes. Also, recall there are $L - t(i) + d(i) - 2$ transit wavelengths, and these are $\{t(i), t(i) + 1, ..., L - t(i) + d(i) - 3\}$. For each transit wavelength, optical signals pass through the hub (without electronic conversion) between exactly one pair of links. Note that the optical pass-through forms a two-hop optical connection. For the other $d(i) - 2$ links, the channels at the wavelength are terminated by the $d(i) - 2$ LTEs at the hub that are dedicated to the wavelength. Note that the star subnetwork has one-hop optical connections at local wavelengths, and at each transit wavelength, has one two-hop optical connection and $d(i) - 2$ one-hop connections.

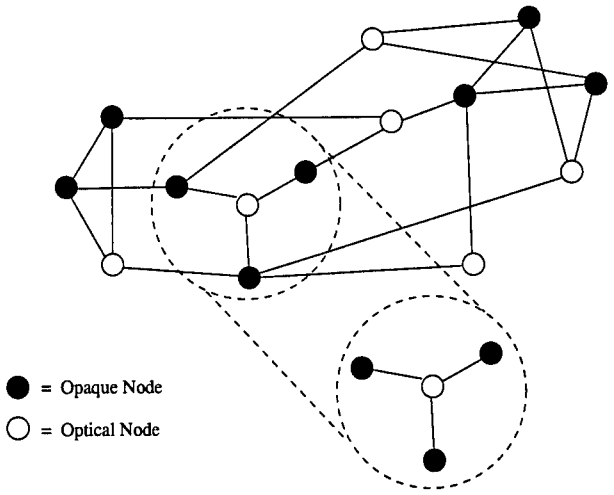



Figure E.6: A star subnetwork with an optical node at the hub.

A traffic stream in the star subnetwork either terminates at the hub or passes through. If it terminates, it is called *local*, and if it passes through, it is called *transit*. Local traffic streams are assigned to local wavelengths. Transit traffic streams are assigned to transit wavelengths first. A transit stream is assigned to any available partially filled two-hop optical connection (following its path) at a transit wavelength. If no such connection exists then such a connection is formed on an available wavelength (recall that a wavelength can only have one such connection). Then the transit stream is assigned to the connection. If no two-hop connection is available then the stream is assigned to one-hop optical connections with a preference for those at transit wavelengths before those at local wavelengths.

Now let us see how the traffic streams fill the channels. Initially, local traffic fill the channels at local wavelengths, and transit traffic fill the channels at transit wavelengths. Note that local traffic will only occupy the channels at local wavelengths because of the node traffic constraint and there are $t(i)$ local wavelengths. Note that transit traffic will first occupy two-hop optical connections at transit wavelengths before occupying one-hop connections. In addition, by the way the traffic streams fill the two-hop connections, there can be at most one partially filled two-hop optical connection over any pair of links.

We will now prove by contradiction that there is no blocking. Suppose a traffic stream is blocked, and let $e_1$ be a link that caused the blocking. Consider the case when the stream is transit. Since the stream is blocked at $e_1$, all one-hop optical connections through $e_1$ are filled, and all two-hop optical connections through $e_1$ are filled or, if partially filled, do not follow the stream's path. There can be at most $d(i) - 2$ partially filled two-hop connections through $e_1$ because over any pair of links, there can be at most

one partially filled two-hop connection. Thus, the number of partially filled channels in $e_1$ is at most $d(i)-2$ and the other channels are completely filled. Since there are $L+d(i)-2$ channels in link $e_1$, the number of traffic in link $e_1$ contradicts the link traffic constraint.

Now consider the case when the stream is local. Then all the channels in $e_1$ at local wavelengths are filled. This implies some transit traffic is using a local wavelength. Since transit traffic has a preference for transit wavelengths, we can use the argument above to conclude that all transit channels are full except for at most $d(i)-2$ partially filled ones. This combined with the fact that all local wavelength channels are filled contradicts the link traffic constraint.

## E.4 A SIMPLE DESIGN ALGORITHM

The cost of a optical network depends on the choice of the subset of optical nodes $O$. In this section, we consider the network design problem to choose the subset $O$ that minimizes (E.1). Therefore, the problem is to find an *independent* set of nodes in the topology that maximizes (E.1), or equivalently that maximizes $\sum_{i\in O} f(i)$, where

$$f(i) = 2(L - t(i)) - (d(i) - 2)(2d(i) - 2). \tag{E.2}$$

The problem is easily shown to be NP-Complete (i.e., inherently difficult). In particular, the NP-Complete problem of finding a maximum-sized independent set in a graph [13] can be transformed (in polynomial time) into the problem of finding an independent subset $O$ that maximizes $\sum_{i\in O} f(i)$. The transformation requires choosing values for $L$ and $t(\cdot)$ so that $f(i) = 1$ for all nodes $i$. (Then maximizing $\sum_{i\in O} f(i)$ means finding a maximum-sized independent set.) The appropriate values of $L$ and $t(\cdot)$ are

$$L = 1 + \max_i \frac{1 + (d(i) - 2)(2d(i) - 2)}{2}$$

and for all nodes $i$,

$$t(i) = L - \frac{1 + (d(i) - 2)(2d(i) - 2)}{2}.$$

We propose a simple heuristic design algorithm that maximizes $\sum_{i\in O} f(i)$. This algorithm is shown in Figure E.7 and we refer to it as the DROP algorithm. The algorithm is an iterative improvement algorithm that starts with some independent subset of nodes $O$ and iteratively increases its size while maintaining its independence property. When the subset $O$ cannot grow anymore the algorithm stops.

The algorithm adds nodes to $O$ using random selection, where nodes that will lead to greater improvement are more likely to be added. Since the algorithm produces a

## DROP ALGORITHM

**Input:** A network topology with nodes $V$, and traffic parameters $L$ and $\{t(i) : i \in V\}$.

**Output:** An independent subset of optical nodes $O$ with large value for $\sum_{i \in O} f(i)$.

(**Definitions:** If $U$ is a subset of $V$ then $f(U) = \sum_{v \in U} f(u)$. A constant $c$ is used in the algorithm which can be tuned to optimize performance.

**Step 1.** Let $O = \emptyset$, the empty set.

**Step 2.** If $O$ is a maximal independent subset then STOP.

**Step 3.** Let $I$ be the subset of nodes in $V \backslash O$ such that if $v \in I$ then $O \cup \{v\}$ is independent.

**Step 4.** For $v \in I$, let $\Delta(v) = f(O \cup \{v\}) - f(O)$.

**Step 5.** Let $T = \frac{\sum_{v \in I} |\Delta(v)|}{|I|}$. If $T = 0$ then STOP. *(Comment: $T$ is the average change in cost by perturbing the subset $O$ by one node, where the average is taken over all possible nodes. Notice if $T = 0$ then we can expect no improvement, and so we stop.)*

**Step 6.** For $v \in I$, let $p(v) = e^{\Delta(v)/(c \cdot T)} / \sum_{u \in I} e^{\Delta(u)/(c \cdot T)}$.

**Step 7.** Randomly choose a node $v$ according to the probabilities $(p(v) : v \in I)$. Let $O = O \cup \{v\}$. Go to Step 2.

Figure E.7: The DROP algorithm.

random solution, it may be run multiple times to get multiple solutions, and the best one can be chosen as a final solution. The algorithm has a parameter $c$ which can be tuned to optimize performance. The parameter determines how likely the algorithm will choose a node that leads to the *greatest* improvement. In particular, if $c$ is very small then the algorithm always chooses a node that leads to the greatest improvement. On the other hand, if $c$ is very large then the algorithm chooses a node without any preference, i.e., each node is equally likely.

DROP has seven steps as shown in Figure E.7. Step 1 initializes the independent set $O$ to the empty set. Steps 2 through 7 is an iteration step of adding a new node to $O$. Step 2 stops the algorithm if $O$ is a maximum sized independent set. In Step 3, all possible nodes that can be added to $O$ are determined. Step 4 computes the change in cost $\Delta(v)$ if node $v$ is added to $O$. In Step 5, a value $T$ is computed which is an average of the change in cost if subset $O$ were perturbed by one node. It will be used as a normalizing constant for change of cost values in the next step. In Step 6, the probabilities of choosing a node based upon the $\Delta(\cdot)$ values. Notice that $\Delta(\cdot)$ is normalized by $T$ and is scaled by the parameter $c$. Finally, Step 7 selects a new node using the probabilities and adds the node to $O$.

The curves shown in Figure E.8 show how the performance of the algorithm is dependent on the parameter $c$ for the NSFNET topology (shown in Figure E.3), $L = 32$, and where the $\{t(i)\}$ are chosen randomly and uniformly over an interval. The curves correspond to the intervals [10..22] and [6..26]. Note that the expected value of $t(i)$ is 16, which is half the total number of wavelengths. Each curve shows the average number of runs of DROP before an optimal solution is found. Each data point corresponds to an average over 400 different sets of $\{t(i)\}$ values. The best value of $c$ turns out to be a fraction around $10^{-.5} = 0.32$ for this topology. Then the average number of runs of DROP before finding the optimal solution is around 5.

## E.5 SUMMARY

We presented a network architecture that allows optical signals to bypass nodes. Assuming that the traffic is incremental and has link and node traffic constraints, the network performs as well as one where all switching is done electronically. We show through an NSFNET example that equipment costs can be reduced significantly with only a small increase in the number of wavelengths. We also provide a simple heuristic to place optical nodes to minimize cost.

**Acknowledgements**

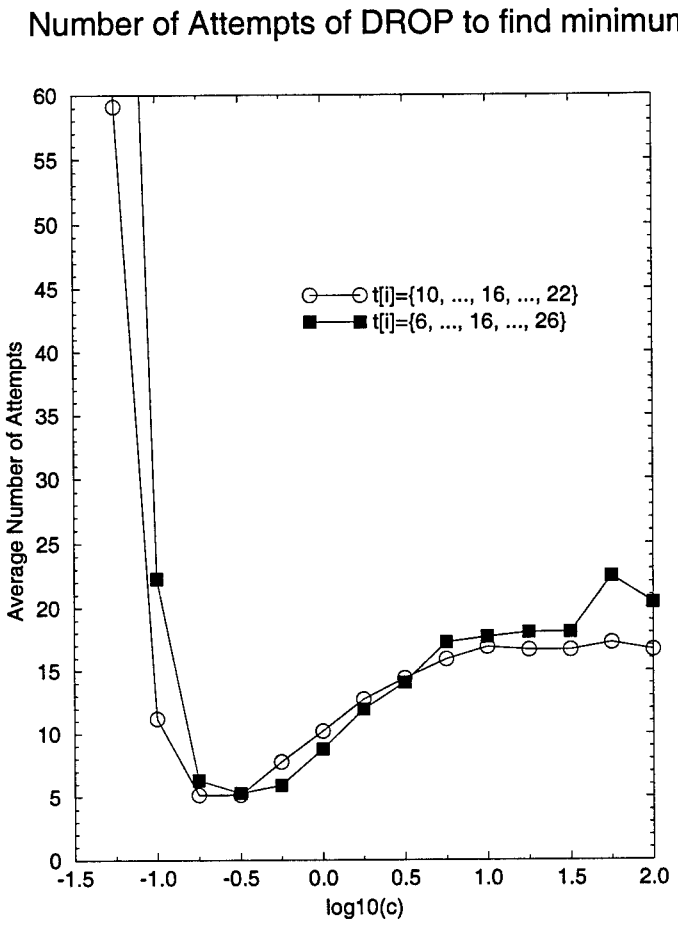



Figure E.8: Sensitivity of DROP with parameter $c$.

Effort sponsored by the Defense Advanced Research Projects Agency (DARPA) and Rome Laboratory, Air Force Materiel Command, USAF, under agreement number F30602-97-0342. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright annotation thereon. The views and conclusions contained therein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Defense Advanced Research Projects Agency (DARPA), Rome Laboratory, or the U.S. Government.

# Appendix F

# List of Symbols, Abbreviations, and Acronyms

**ADM:** Add-drop multiplexer.

**Brute Force Search:** A channel assignment algorithm that assumes that lightpath requests do not come with pre-computed routes.

**CAP:** Channel attachment pattern.

**DCS:** Digital cross-connect system.

**DesignCAP:** Algorithm that designs CAPs for optimal performance.

**DFS:** Depth First Search channel assignment algorithm

**DISTRIBUTE:** A local CAP.

**E:** Usually means number of links.

**Fixed:** A wavelength preference that depends on the wavelength index.

**g:** The number of tributary traffic streams that fit into a lightpath.

**LTE:** Line terminating equipment.

**Multiple-MTV:** Rearranges one or more lightpaths in MTV fashion.

**MTV:** Move-to-vacant lightpath rearrangement.

**N:** Usually means number of nodes.

**OXC:** Optical cross-connect.

**Pack:** A wavelength preference that depends on the utilization of wavelengths.

**PARTITION:** A local CAP.

**Random:** A wavelength preference that is random.

**RANDOM-RANDOM:** A random CAP that is different for each node.

**RANDOM-SAME:** A random CAP that is the same for each node.

**Single-MTV:** Rearranges one lightpath in MTV fashion.

**Shortest Distance:** A channel assignment algorithm that assumes that lightpath requests do not come with pre-computed routes.

**SP:** Shortest Path, which is a channel assignment algorithm that assumes lightpath requests come with pre-computed routes.

**SHIFTED:** A CAP that is modified from PARTITION.

**SHUFFLE:** A CAP that spreads the w-attachments.

**Spread:** A wavelength preference that depends on the utilization of wavelengths.

**RAS:** Rearrangement after shut-down.

**Widest Shortest:** A channel assignment algorithm that assumes that lightpath requests do not come with pre-computed routes.

**W:** Number of wavelengths.

**WDM:** Wavelength division multiplexing.

*U.S. GOVERNMENT PRINTING OFFICE: 2000-510-079-10010

# DISTRIBUTION LIST

| addresses | number of copies |
|---|---|
| AFRL/IFGA<br>ATTN: PAUL SIERAK<br>525 BROOKS ROAD<br>ROME, NEW YORK 13441-4505 | 2 |
| UNIVERSITY OF HAWAII<br>ATTN: DR. SASAKI<br>DEPT OF EE<br>2540 DOLE STREET<br>HONOLULU HI 96822 | 5 |
| AFRL/IFOIL<br>TECHNICAL LIBRARY<br>26 ELECTRONIC PKY<br>ROME NY 13441-4514 | 1 |
| ATTENTION: DTIC-OCC<br>DEFENSE TECHNICAL INFO CENTER<br>8725 JOHN J. KINGMAN ROAD, STE 0944<br>FT. BELVOIR, VA 22060-6218 | 1 |
| DEFENSE ADVANCED RESEARCH<br>PROJECTS AGENCY<br>3701 NORTH FAIRFAX DRIVE<br>ARLINGTON VA 22203-1714 | 1 |
| ATTN: NAN PFRIMMER<br>IIT RESEARCH INSTITUTE<br>201 MILL ST.<br>ROME, NY 13440 | 1 |
| AFIT ACADEMIC LIBRARY<br>AFIT/LDR, 2950 P.STREET<br>AREA B, BLDG 642<br>WRIGHT-PATTERSON AFB OH 45433-7765 | 1 |
| AFRL/MLME<br>2977 P STREET, STE 6<br>WRIGHT-PATTERSON AFB OH 45433-7739 | 1 |

AFRL/HESC-TDC
2698 G STREET, BLDG 190
WRIGHT-PATTERSON AFB OH 45433-7604 — 1

ATTN: SMDC IM PL
US ARMY SPACE & MISSILE DEF CMD
P.O. BOX 1500
HUNTSVILLE AL 35807-3801 — 1

TECHNICAL LIBRARY D0274(PL-TS)
SPAWARSYSCEN
53560 HULL ST.
SAN DIEGO CA 92152-5001 — 1

CDR, US ARMY AVIATION & MISSILE CMD
REDSTONE SCIENTIFIC INFORMATION CTR
ATTN: AMSAM-RD-OB-R, (DOCUMENTS)
REDSTONE ARSENAL AL 35898-5000 — 2

REPORT LIBRARY
MS P364
LOS ALAMOS NATIONAL LABORATORY
LOS ALAMOS NM 87545 — 1

ATTN: D'BORAH HART
AVIATION BRANCH SVC 122.10
FOB10A, RM 931
800 INDEPENDENCE AVE, SW
WASHINGTON DC 20591 — 1

AFIWC/MSY
102 HALL BLVD, STE 315
SAN ANTONIO TX 78243-7016 — 1

ATTN: KAROLA M. YOURISON
SOFTWARE ENGINEERING INSTITUTE
4500 FIFTH AVENUE
PITTSBURGH PA 15213 — 1

USAF/AIR FORCE RESEARCH LABORATORY
AFRL/VSOSA(LIBRARY-BLDG 1103)
5 WRIGHT DRIVE
HANSCOM AFB MA 01731-3004 — 1

ATTN: EILEEN LADUKE/D460 1
MITRE CORPORATION
202 BURLINGTON RD
BEDFORD MA 01730

OUSD(P)/DTSA/DUTD 1
ATTN: PATRICK G. SULLIVAN, JR.
400 ARMY NAVY DRIVE
SUITE 300
ARLINGTON VA 22202

# *MISSION*
# *OF*
# *AFRL/INFORMATION DIRECTORATE (IF)*

The advancement and application of information systems science and technology for aerospace command and control and its transition to air, space, and ground systems to meet customer needs in the areas of Global Awareness, Dynamic Planning and Execution, and Global Information Exchange is the focus of this AFRL organization. The directorate's areas of investigation include a broad spectrum of information and fusion, communication, collaborative environment and modeling and simulation, defensive information warfare, and intelligent information systems technologies.